

PA - Personnel Management: Workflow Scenarios



HELP.BCBMTWFMPA

Release 4.6C



Copyright

© Copyright 2001 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® are registered trademarks of Microsoft Corporation.

IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.

ORACLE® is a registered trademark of ORACLE Corporation.

INFORMIX®-OnLine for SAP and Informix® Dynamic Server™ are registered trademarks of Informix Software Incorporated.

UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.






HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA® is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Contents

PA - Personnel Management: Workflow Scenarios	9
SAP Business Workflow for Recruitment (PA-RC)	9
Receipt and Resubmission	9
Graphic: Receipt/Resubmission	10
Planning and Holding Job Interviews	11
Graphic: Job Interview	12
Creating an Offer of Contract	13
Graphic: Offer of Contract	14
Preparation for Hiring	15
Graphic: Preparation for Hiring	15
Management of Applicants on Hold	16
Graphic: Monitoring Applicants On Hold	17
Rejecting Applicants	17
Technical Background (PA-RC)	18
Applicant: Applicant	18
Application: Applicant Application	21
Display Applications for an Applicant	24
Applicant Document: Applicant Document	24
Applicant Activity: Applicant Activity	24
Display Applicant Activity	26
Change Applicant Activity	27
Vacancy Assignment: Applicant Vacancy Assignment	28
Preparation and Customizing (PA-RC)	30
Specific Customizing Using Examples	31
Parameter Settings for Applicant Activities	31
Setting Parameters for Applicant Activities	32
New Standard Text for Rejection Letter: Example	33
Differentiating Between Applicant Correspondence	34
Differentiating Between Applicant Correspondence	34
Differentiating Between Rejection Letters: Example	35
Adding New Tasks	37
Adding New Tasks	37
New Appl. Act: Appointment for Apt. Test: Example	38
Modification of Deadline Monitoring	39
Modifying Deadline Monitoring	40
Modifying Applicants' Length of Time in Appl. Pool	41
Using and Linking to Application Functions (PA-RC)	41
Start/End of Workflow	41
Controlling the Workflow Using Applicant Status	41
Controlling the Workflow Using Vacancy Assignment	42
Use of Applicant Activities in Workflow	43
Workflow Scenarios in Pers. Administration (PA-PA)	44
Follow-Up Activities For Hiring (PA-PA)	44
Using and Linking to Application Functions (PA-PA)	45

Workflow	46
ESS - Follow-Up Activities For a New Hire (PA-PA).....	46
Using and Linking to Application Functions (PA-PA).....	47
Delimiting an ESS User (PA-PA).....	48
Using and Linking to Application Functions (PA-PA).....	49
Approving an Individual BP Increase (PA-PA) (Obs.)	49
Using and Linking to App. Functions (PA-PA) (Obs.)	50
Process Leave Request (PA-TIM-REC)	52
Using & Linking to Functions (PA-TIM-REC).....	53
Preparation and Customizing (PA-TIM-REC).....	54
Relate System User Name to Personnel Number	54
Set Up The Organizational Plan.....	55
Cancel Leave Request (PA_TIM_REC).....	55
Using & Linking to Functions (PA-TIM-REC).....	56
Preparation and Customizing (PA-TIM-REC).....	57
Relate System User Name to Personnel Number	57
Set Up The Organizational Plan.....	58
Workflow Scenarios in Travel Management (FI-TV)	58
Approve Travel Request.....	59
Graphic: Approve Travel Request.....	59
Approve Travel Plan.....	60
Graphic: Approve Travel Plan	62
Approve Trip	64
Graphic: Approve Trip	65
Technical Background (FI-TV)	66
Object Type BUS2089 (Employee Trip).....	68
OT SELFITEM (Work Item for Dialog Steps on Self).....	68
Standard Task TS20000118 (Approve Travel Request).....	69
Standard Task TS20000130 (Set Trip Status to Approved)	69
Standard Task TS20000095 (Send Mail: Request Approved).....	70
Standard Task TS01000124 (Approve Travel Plan)	70
Standard Task TS01000126 (Send Mail: Travel Plan Approved).....	71
Standard Task TS60300084 (Change Status of Travel Plan)	71
Standard Task TS60300085 (Write PNR to Processing Queue).....	72
Standard Task TS01000129 (Change Travel Plan).....	72
Standard Task TS60300086 (Cancel Travel Plan)	73
Standard Task TS20000189 (Change Trip).....	73
Standard Task TS20000139 (Enter and Send Short Message)	74
Standard Task TS20000135 (Check Trip Facts)	74
Standard Task TS20000131 (Approve Trip Facts)	75
Template WS20000050 (Approve Travel Request).....	75
Workflow Template WS 01000087 (Approve Travel Plan)	76
Workflow Template WS20000040 (Approve Trip)	77
Preparation and Customizing (FI-TV).....	78
Relationship Between System User Name & Pers. No.	78
Defining the Organizational Plan.....	79
Task-Specific Customizing.....	79
Maintenance of Event Receiver Linkage	80
Using and Linking to Application Functions (FI-TV)	80

Automatic Generation of Documents for HR Applications	81
Design Implementation in SAP Business Workflow®	82
Technical Background	83
Business Objects	83
Event Generation	84
HR Status Management	84
Define Standard Tasks	85
Object Method	86
Container Elements	86
Role	87
Example: Standard Task for Printing a Contract	87
Workflow Definition	88
Workflow Editor	89
Workflow Container	89
Triggering Events	89
Workflow Steps	90
Binding Between Workflow and Standard Task	91
Responsibility	91
Exceptions	92
Using and Linking to Application Functions	92
Create Documents Using SAPscript	93
Define Main Text	93
Define Form	94
Use Container Element Data in Form Windows	95
Printing a P46 Form (Company Cars) (PA-PAY-GB)	96
Technical Background (PA-PAY-GB)	96
Object Type BUSMILEAGE (Business Mileage)	97
Object Type CARCAPACIT (Car Capacity)	97
Object Type COMPCARGB (Company Car GB)	97
Object Type NINSURANCE (National Insurance)	98
Object Type PROLLCAR (Payroll Car)	98
Standard Task TS00408216 (HRGBBusLimit)	98
Standard Task TS00408217 (HRGBHubraum)	99
Standard Task TS00408218 (HRGBRegdate)	100
Standard Task TS20000020 (HRGBPrivPay)	100
Standard Task TS20000042 (HRGBConvert)	101
Workflow Template WS00400251 (HR_GB:PrtCar)	102
Event Generation	104
Maintenance of Event Receiver Linkage	104
Operation and Connection to Application Functionality (PA-PAY-GB)	105
Workflow Scenarios in Compensation Management (PA-CM)	106
Compensation Adjustment Approval Process (CMPApprProc)	106
Approve and Activate Adjustments	108
Maintenance of Event Receiver Linkage	110
Follow-up Actions After Award Exercised (CMPEexerc)	110
Exercising Awards	111
Award Expiration (CMPExpire)	112
Notification of Expiration of Awards	113

Workflow Scenarios in Personnel Development (PA-PD)	114
Qualification Monitoring (PA-PD)	114
Technical Implementation (PA-PD)	116
Object Type BUS7019 (PA-PD)	116
Standard Task PDProfileMat (PA-PD)	117
Role of Personnel Administrator (PA-PD)	117
Workflow Template PDExpiredQua (PA-PD)	118
Preparations and Customizing (PA-PD)	120
Working Through Task-Specific Customizing	121
Linking Events to Event Receivers	121
Connection to Application Functions (PA-PD)	121
Approving and Following Up an Appraisal (PA-PD)	122
Technical Implementation (PA-PD)	124
Object Type Appraisal (PA-PD)	125
Standard Task PdvApprove (PA-PD)	126
Standard Task PdvReject (PA-PD)	127
Standard Task PdvApprDecis (PA-PD)	127
Standard Task PdvStatus (PA-PD)	128
Standard Task PdvChangAppr (PA-PD)	128
Standard Task PdvAssesBack (PA-PD)	129
Standard Task PdvAssesDia (PA-PD)	130
Role of Appraiser (PA-PD)	130
Role of Supervisor of Appraisee (PA-PD)	131
Role of Appraisee (PA-PD)	132
Workflow Template PdvAppraisal (PA-PD)	132
Workflow Definition Details	133
Workflow Template PdvAppEmpl (PA-PD)	136
Preparations and Customizing (PA-PD)	136
Working Through Task-Specific Customizing	137
Linking Events to Event Receivers	137
Connection to Application Functions (PA-PD)	138
Changed Subprofile (PA-PD)	139
Technical Implementation (PA-PD)	141
Object Type BUS7030 (PA-PD)	141
Standard Task PdvDispProfi (PA-PD)	142
Role of Employee (PA-PD)	142
Workflow Template PDVChanProfi (PA-PD)	143
Preparations and Customizing (PA-PD)	145
Working Through Task-Specific Customizing	145
Linking Events to Event Receivers	146
Profile Matchup for Transfer to Another Position (PA-PD)	146
Technical Implementation (PA-PD)	147
Object Type PDOTYPE_SH (PA-PD)	148
Standard Task PdvPosTransf (PA-PD)	148
Role of Employee (PA-PD)	149
Workflow Template PdvTransfer (PA-PD)	150

Preparations and Customizing (PA-PD)	152
Working Through Task-Specific Customizing	152
Linking Events to Event Receivers	152

PA - Personnel Management: Workflow Scenarios

See also:

[BC - Workflow Scenarios in Applications \[Ext.\]](#)

SAP Business Workflow for Recruitment (PA-RC)

Purpose

The recruitment workflow supports the entire recruitment procedure, from initial entry of applicant data to hiring or rejecting an applicant.

The standard system contains a recruitment workflow scenario that consists of six workflow templates:

- Applicant processing (main workflow)
 - This workflow is started for each application recorded in the system. It creates the required confirmations of receipt and letters of rejection. It controls and coordinates the following sub-workflows:
 - [Receipt and Resubmission \[Page 9\]](#)
 - [Planning and Holding Job Interviews \[Page 11\]](#)
 - [Creating an Offer of Contract \[Page 13\]](#)
 - [Preparation for Hiring \[Page 15\]](#)
 - [Management of Applicants on Hold \[Page 16\]](#)

Receipt and Resubmission

The workflow template for receipt and resubmission represents the following process:

After an applicant has been recorded in the system, the personnel officer responsible must decide on the further procedure for the applicant. The personnel officer can assign applicants to vacant positions, reject them, or put them on hold.

If an applicant appears to be suitable for a vacant position, the personnel officer assigns him or her to the position in question. The person responsible for the position must then decide whether the applicant is to be invited for an interview, or rejected.

If the person responsible for the position decides to invite the applicant for an interview, this triggers the [planning and holding of job interviews \[Page 11\]](#).

Graphic: Receipt/Resubmission

If the person responsible for the position is not interested in the applicant, the applicant is rejected. This returns the applicant to *receipt and resubmission*. The personnel officer then decides on the further course of action with regard to the applicant.

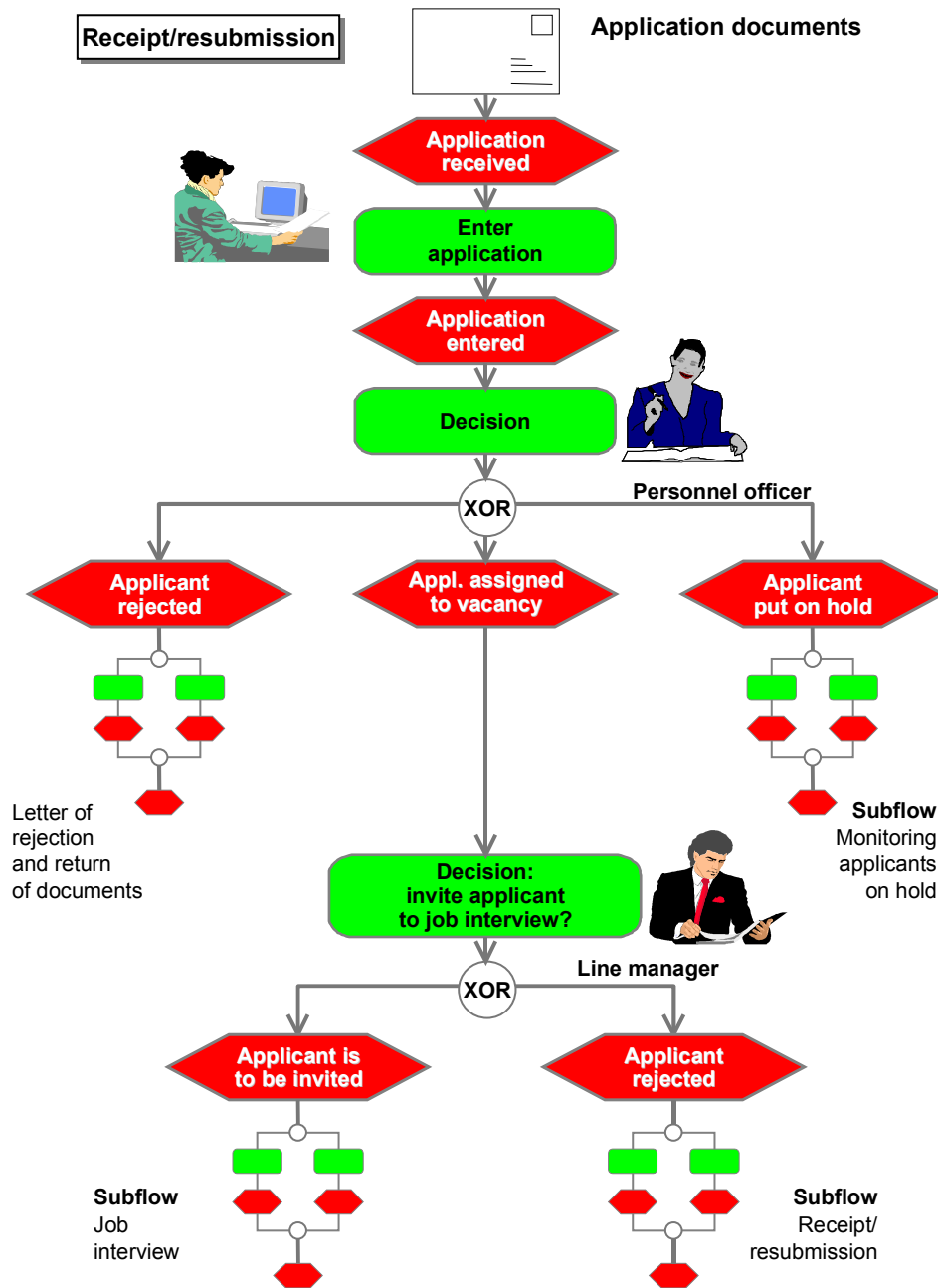
[Graphic: Receipt and Resubmission \[Page 10\]](#)

See also:

[Reject Applicants \[Page 17\]](#)

Graphic: Receipt/Resubmission

Planning and Holding Job Interviews



Planning and Holding Job Interviews

The workflow template for planning and holding job interviews represents the following process:

Graphic: Job Interview

If an applicant is to be invited for a job interview, the person responsible for the position proposes a date and time, together with any possible alternatives. He or she also asks the personnel officer to arrange an appointment with the applicant.

The personnel officer arranges the appointment for a job interview. If necessary, he or she changes the date and time proposed by the person responsible for the position.

Triggered by the decision to invite the applicant for a job interview, the system assigns the applicant a letter of invitation. This letter can be printed automatically in bulk processing, or individually by the personnel officer. The letter contains the data stored on the job interview. If the personnel officer prints an individual letter, he or she must confirm that it has been sent to the applicant.

At the start of the interview, it is the personnel officer's task to create a log of the job interview in the system. After this log has been created, he or she must flag the task as completed.

After the job interview, the person responsible for the position must decide whether the applicant is to be offered a contract, or rejected.

If he or she decides that the applicant is to be offered a contract, the personnel officer sets the applicant's overall status to *Contract offered*. This triggers the [creation of an offer of contract \[Page 13\]](#).

If the person responsible for the position rejects the applicant, the applicant is returned to [receipt/resubmission \[Page 9\]](#). The personnel officer then decides on the further course of action with regard to the applicant.

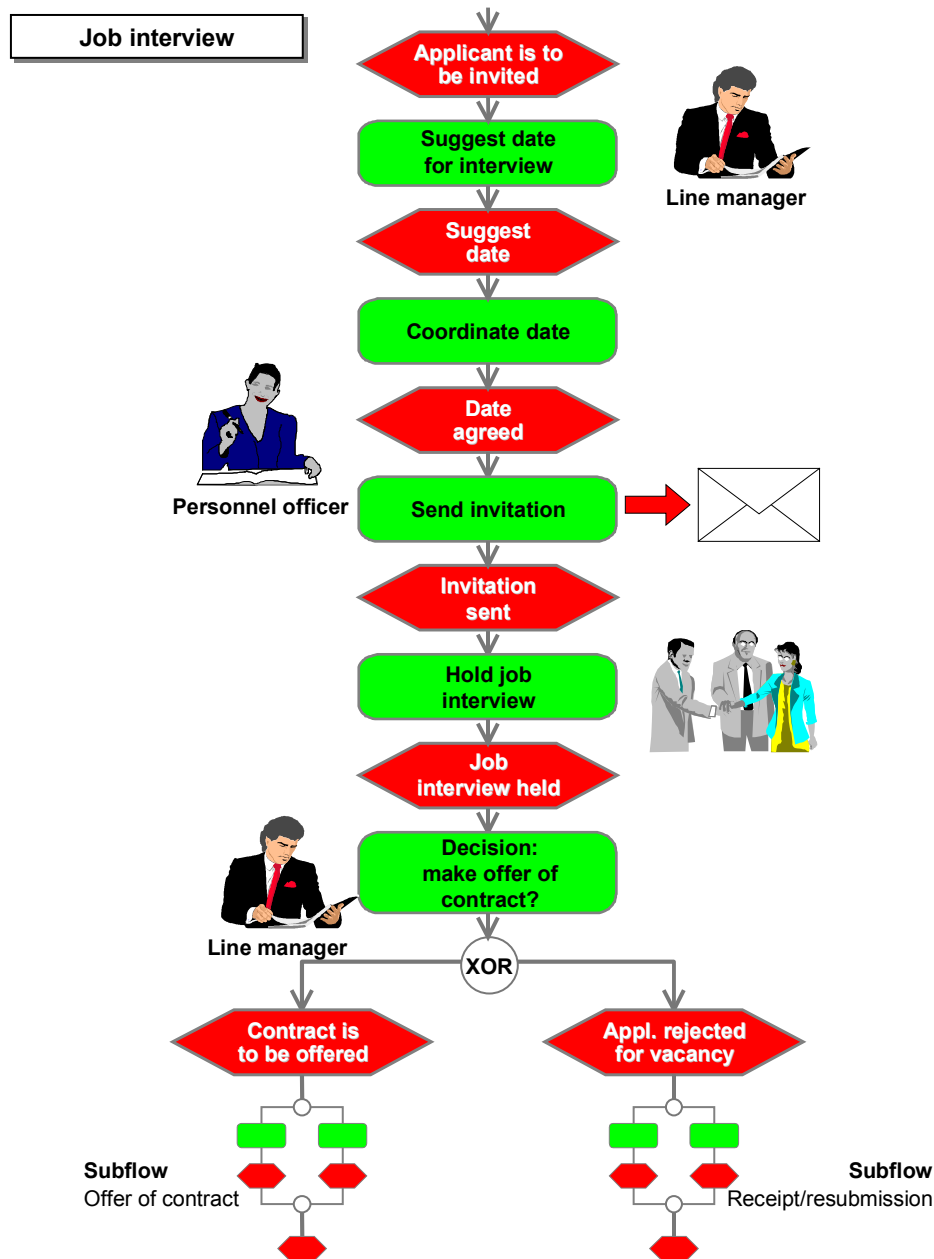
[Graphic: Job Interview \[Page 12\]](#)

See also:

[Reject Applicants \[Page 17\]](#)

Graphic: Job Interview

Creating an Offer of Contract



Creating an Offer of Contract

The workflow template used to create an offer of contract describes the following process:

If an applicant is assigned the overall *Contract offered* status, the personnel officer creates an employment contract for the applicant. The appropriate manager checks the employment contract, and can add any missing data if necessary.

Graphic: Offer of Contract

The appropriate manager is sent a reminder to sign the employment contract. The fact that the employment contract has been signed must be confirmed in the system.

The personnel officer is then requested to mail the employment contract, and confirm that he or she has done so.

The applicant must return the signed employment contract within a period determined in workflow. After this period has elapsed, the personnel officer receives a reminder.

If the contract is received by the date specified, the personnel officer changes the applicant's overall status to *To be hired*. This triggers [preparations for hiring \[Page 15\]](#).

If the deadline is not met, he or she sets the applicant's overall status to *Rejected*.

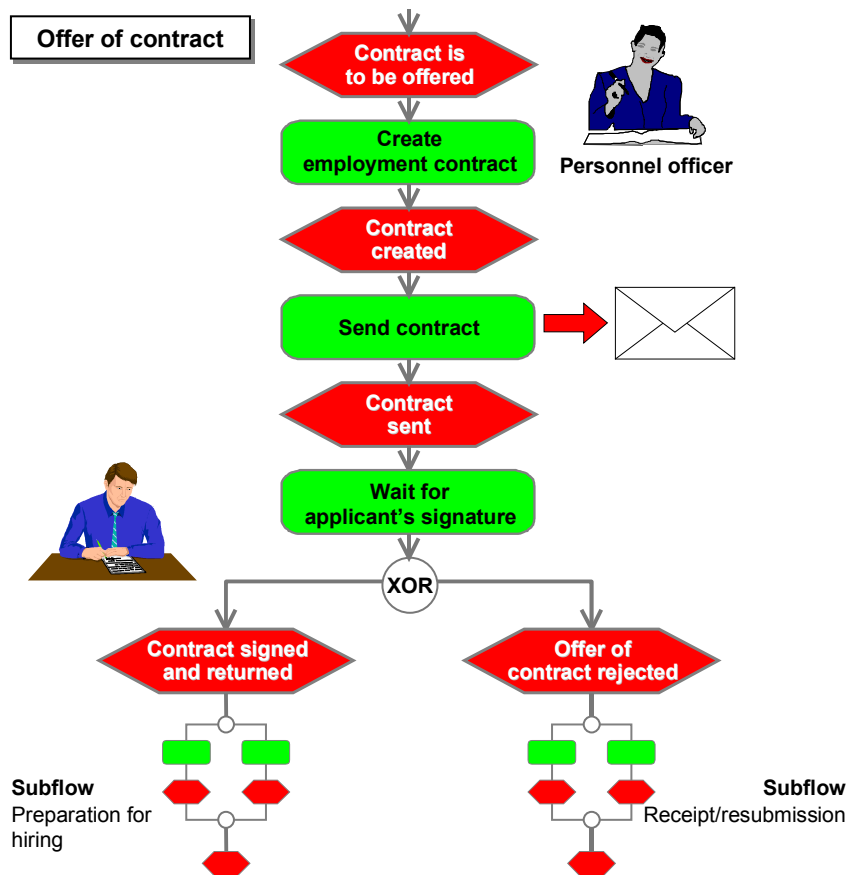
[Graphic: Offer of Contract \[Page 14\]](#)

See also:

[Reject Applicants \[Page 17\]](#)

Graphic: Offer of Contract

Preparation for Hiring



Preparation for Hiring

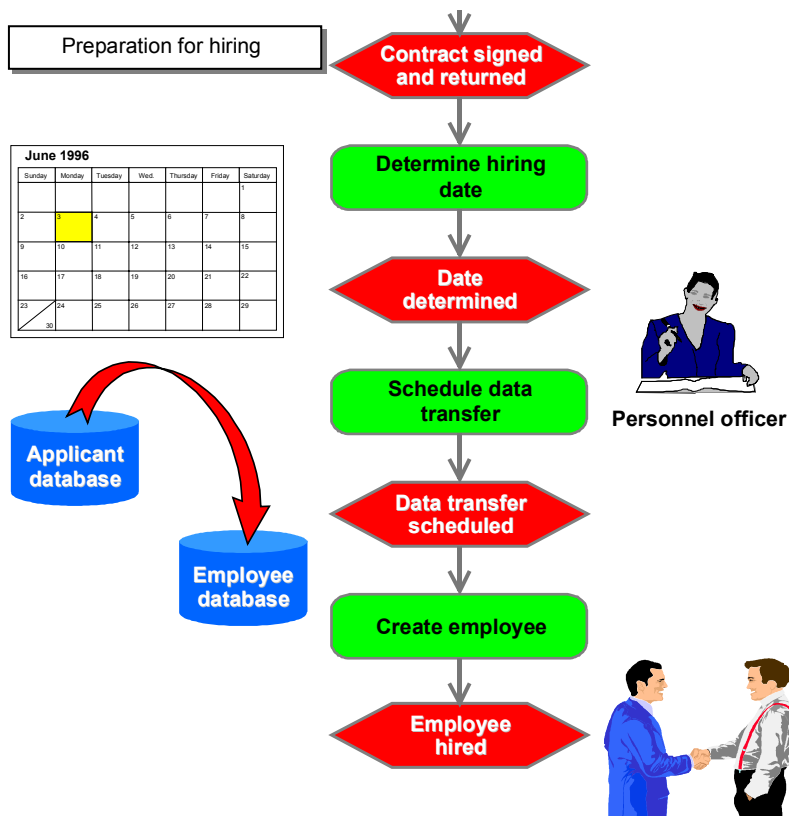
The workflow template used to prepare for hiring represents the following process:

If an applicant is assigned the overall *To be hired* status, the personnel officer responsible must specify the required hiring date. The personnel officer is then required to transfer the applicant's data to *Personnel Administration*.

[Graphic: Preparation for Hiring \[Page 15\]](#)

Graphic: Preparation for Hiring

Management of Applicants on Hold



Management of Applicants on Hold

The workflow template for managing applicants on hold represents the following process:

If an applicant is assigned the *On hold* overall status, a deadline is set to monitor the length of time for which the applicant is kept on hold.

If the applicant is not assigned to a new vacancy before the deadline, he or she is rejected when the deadline is reached.

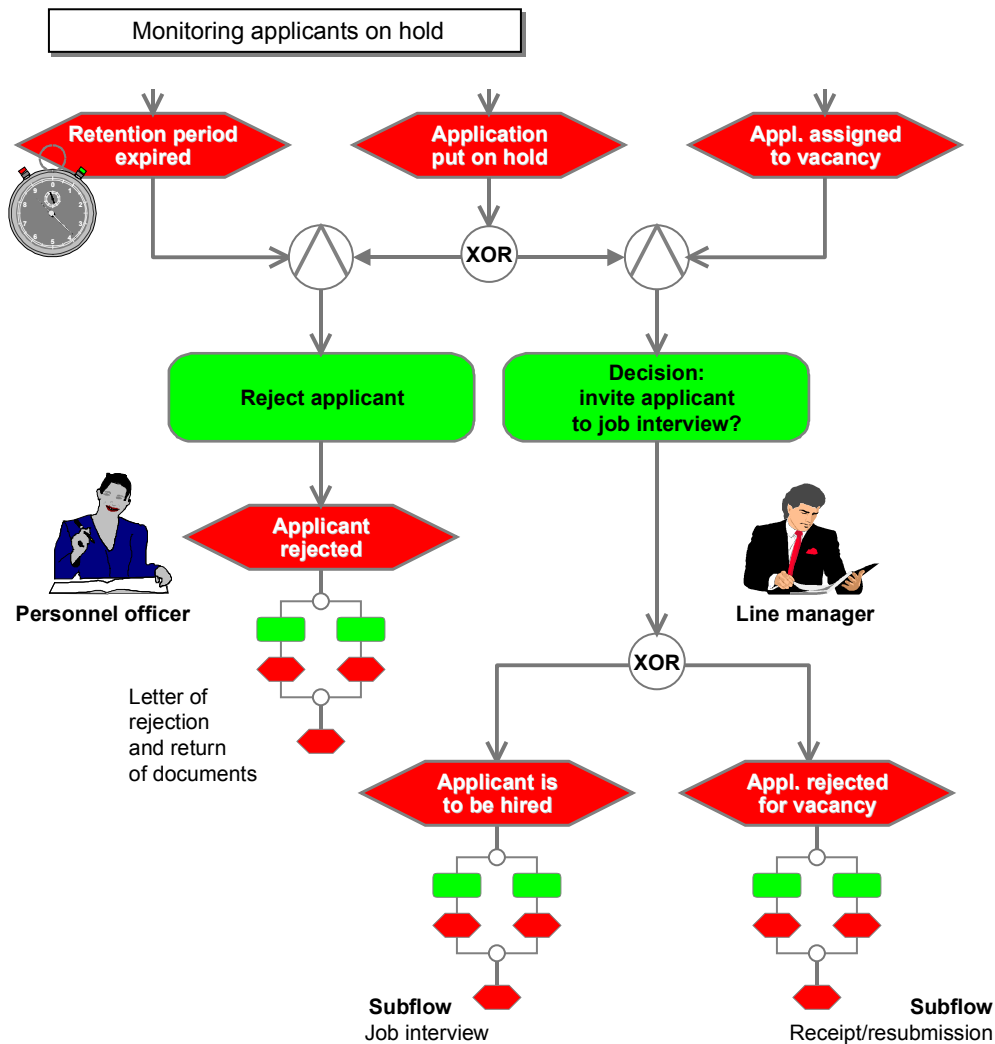
If an applicant on hold is selected before the deadline is reached, monitoring is canceled.

[Graphic: Management of Applicants on Hold \[Page 17\]](#)

See also:

[Reject Applicants \[Page 17\]](#)

Graphic: Monitoring Applicants On Hold



Rejecting Applicants

Rejected applicants are always processed immediately, irrespective of the time or situation in which the rejection takes place.

If an applicant is rejected for a vacancy, he or she is returned to [receipt/resubmission \[Page 9\]](#). The personnel officer then decides on the further course of action with regard to the applicant.

If an applicant is rejected completely, the system creates a letter of rejection in background processing. After the personnel officer has confirmed that this letter has been mailed, the workflow is complete.

Technical Background (PA-RC)

Description of Object Types

The interface between R/3 and the Workflow System is based on a structure of objects.

The following information is of a technical nature. You require this information if you are interested in implementation details or want to perform your own enhancements.

The standard system contains the following object types:

[Applicant: Applicant \[Page 18\]](#)

[Application: Applicant Application \[Page 21\]](#)

[ApplicantDocument: Applicant Document \[Page 24\]](#)

[ApplicantActivity: Applicant Activity \[Page 24\]](#)

[VacancyAssignment: Applicant Vacancy Assignment \[Page 28\]](#)

Applicant: Applicant

The applicant's business application object is represented by the *Applicant* object type. The applicant number serves as a unique key for this object type.

Attributes

When the workflow is run, the following fields are available as attributes:

- *Personnel number of internal or external applicants*
- *Internal or external applicant*
- *Name of applicant*
- *Overall status of applicant*
- *Applicant group*
- *Applicant range*
- *Reason for overall status*
- *Language key for applicant correspondence*
- *Personnel officer*
- *Applicant number and name of applicant*

Applicant: Applicant

- *Vacancy ID of highest valid assignment.* If there are several vacancy assignments, this is the one with the highest priority that also has the 'processing' status.

The following objects are also available as attributes:

- *Personnel (sub)area*
- *Vacancy assignments*
- *Highest valid vacancy assignment.* If there are several vacancy assignments, this is the one with the highest priority that also has the 'processing' status.
- *Applications of an applicant*
- *Personnel officer (applicant)*

Events

The *Applicant* object type can have the following statuses:

- *Applicant created*
- *Application created for applicant*
- *Vacancy assignment created for applicant*

Each change of status can be included in the Workflow as an event.

Moreover, each change to an applicant's overall status can be included in the Workflow as an event (for example, applicant put on hold, contract offered, applicant rejected).

Object Methods

The following methods can be used for the *Applicant* object type:

- **Create (Applicant.Create)**
- **Display (Applicant.Display)**
- **Change (Applicant.Edit)**
- **Create Applicant Action (Applicant.ActionCreate)**

Create (Applicant.Create)

The *Create* object method corresponds to the [Initial Entry of Basic Data \[Ext.\]](#) transaction in the application. Entering an applicant's basic data creates the applicant in the system.

Display (Applicant.Display)

The *Display* object method corresponds to the *short profile* application function. It lists the most important data on the applicant, such as personal data and information on training, qualifications, etc. You can use the short profile to help you reach a decision when [making your applicant selection \[Ext.\]](#), for example.

Change (Applicant.Edit)

Data on an applicant is grouped in infotypes. After an applicant's data has been recorded using the *Initial entry of basic data* personnel action, you may need to create additional data (such as the applicant's bank details) or change existing data.

Applicant: Applicant

Data maintenance is commonly required for a change of address, or to delete an applicant's secondary residence, for example.

The *Change* object method is a synchronous method that can be performed online.

The *Change* object method uses the following parameters:

- *FunctionCode* = MOD (maintain single screen), LIS1 (maintain list screen)
- *Infotype* = Infotype
- *Subtype* = Subtype
- *BeginDate* = Start of infotype record
- *EndDate* = End of infotype record

You must specify a function code and an infotype key.

This method can be used to maintain all applicant data, provided that the parameters have been set accordingly.



In the standard scenarios of the Recruitment System, for example, the *Change* method is used to change an applicant's vacancy assignment.

The *Change* object method is used in the following standard tasks:

- TS00008147 *Change vacancy assignment*
- TS00200279 *Change applicant data*

They are not used in any of the standard workflow templates.

Create Applicant Action (Applicant.ActionCreate)

The term *applicant action* refers to:

- Process flows that change the applicant's overall status
- Process flows for data entry (for example, initial entry of applicant data, enter additional data)

All actions carried out for an applicant are recorded in infotype 4000 *Applicant Actions*.

The *Create applicant action* object method is a synchronous method that is performed online.

This object method uses the following parameters:

- *Action* = Action key
- *StatusReason* = Action reason
- *SupDg* = Indicates whether the action is performed online

You must specify the action key.

This method can be used in tasks that involve performing different applicant actions, provided that the parameters are set accordingly. You must specify the action key to determine which action is performed.



In the standard scenarios in the Recruitment System, the *Create applicant action* object method is used to change an applicant's overall status, for example, or to enter additional data on an applicant.

The *Create applicant action* object method is used in the following standard tasks:

- TS00008248 *Create applicant action*
- TS00008148 *Reject applicant*
- TS00008247 *Change applicant status to hired*
- TS00008250 *Enter additional data*
- TS00008251 *Put applicant on hold*

Application: Applicant Application

A record is created in infotype 4001 *Applications* for each application received from an applicant. In addition to the date on which the application was received, this record also indicates whether the application refers to an advertisement or was unsolicited.

The application's business application object is represented by the *Application* object type. This object is uniquely identified by the applicant number, the date of receipt, and the sequence number of the application.

Attributes

When the workflow is run, the following fields are available as attributes:

- *Advertisement*
- *Unsolicited applicant group*
- *Applicant activities*

The following objects are also available as attributes:

- *Application* (aggregate object)

Events

The *Application* object type can have the following statuses:

- *New application created*

Each change of status can be included in the Workflow as an event.

Object methods

The following methods can be used for the *Application* object type:

- **Display (Application.Display)**

Application: Applicant Application

- **Create Applicant Activity Without Dialog (Application.ActivityCreateAutom)**
- **Create Applicant Activity With Dialog (Application.ActivityCreate)**



The last two object methods for the *Application* object type refer to applicant activities. Applicant activities are always created for one application. Individual applicants can be assigned different applicant activities, depending on how the application procedure is organized within a company and on the company's interest in an applicant.

Activities can be created with or without dialog.

For further information, please refer to the [Administration of Selection Processes \[Ext.\]](#) section in the *Recruitment* documentation.

Display (Application.Display)

The *Create* object method corresponds to the [Display Applications for an Applicant \[Page 24\]](#) transaction in the application. The system displays the single screen for the appropriate record of infotype 4001 *Applications*.

Create Applicant Activity Without Dialog (Application.ActivityCreateAutom)

The *Create applicant activity without dialog* object method is always used when all data on the applicant activity is known from the context (for example, when creating letters such as *Confirmation of receipt*) and no additional data needs to be entered.

The *Create applicant activity without dialog* object method is asynchronous, that is, it ends with one of the following events from the application:

- *Applicant activity created*
- *System message generated* (if the applicant activity was not created because of an error)



If an applicant activity cannot be created because of an error, the *System message generated* event can call the *Create applicant activity with dialog* object method. If an error has occurred, the applicant activity is made available in dialog, enabling you to add or correct data.

This function is available in workflow templates WS00400019 and WS00400020. Please use one of these templates when creating activities without dialog.

This object method has the following parameters:

- *IndicCompPlann* = Indicates whether the applicant activity is created as completed or planned
- *Activity* = Activity type
- *ObjectTypeVacancy* = Object type for vacancy assignment
- *Vacancy* = Vacancy assignment
- *StandardText* = Name of text for a standard document

Application: Applicant Application

- *RecurringTask* = Indicates whether an applicant activity must be completed using the recurring tasks function
- *PerformanceDate* = Date for performing the applicant activity
- *Time* = Time for performing the applicant activity
- *ReferenceActivityID* = Indicator for reference activity

You must set the planned/completed indicator, and you must specify the activity type.

For the above reasons (processing when errors occur), the *Create applicant activity without dialog* object method is not used in any standard tasks. Please use the above workflow templates to create applicant activities in the background.

Create Applicant Activity With Dialog (Application.ActivityCreate)

The *Create applicant activity with dialog* object method is used in the following cases:

- Adding or maintaining data on an applicant activity
- Changing a standard letter
- Maintaining notes on an applicant activity
- Setting an applicant activity to completed

The *Create applicant activity with dialog* object method is asynchronous, that is, it ends with the *Applicant activity created* event from the application.



In the standard scenarios in the Recruitment System, the *Create applicant activity with dialog* object method is used to create the *Appointment for interview* applicant activity in dialog, for example. You can specify the performance date, a time and, if required, a building and room number.

This object method has the following parameters:

- *IndicCompPlann* = Indicates whether the applicant activity is created as completed or planned
- *Activity* = Activity type
- *ObjectTypeVacancy* = Object type for vacancy assignment
- *Vacancy* = Vacancy assignment
- *StandardText* = Name of text for a standard document
- *RecurringTask* = Indicates whether an applicant activity must be completed using the recurring tasks function
- *PerformanceDate* = Date for performing the applicant activity
- *Time* = Time for performing the applicant activity
- *ReferenceActivityID* = Indicator for reference activity

You must set the planned/completed indicator, and you must specify the activity type.

The *Create applicant activity with dialog* object method is used in the following standard tasks:

Display Applications for an Applicant

- TS00008244 *Create interview appointment*
- TS00407921 *Create hiring date*
- TS00407873 *Create activity after error message*

Display Applications for an Applicant

1. Choose *App.master data* → *Display*.
The *Display Applicant Master Data* screen appears.
2. In the *Applicant number* field, enter the applicant number that you require.
3. Select the *Applications* infotype, and choose *Display*.
The *Display Applications (Infotype 4001)* screen appears.
4. The *Next record* and *Previous record* functions enable you to scroll between the infotype records for the individual applications.

Result

The applicant's applications are displayed.

ApplicantDocument: Applicant Document

The Recruitment System enables you to store facsimiles of an applicant's documents (for example, letter of application, resume, certificates) as an optical archive, and then display them on screen as required.

The business application object of the archived facsimile of the applicant document is represented by the *Applicant document* object type.

The *Applicant document* object type is a special case.

The methods used for the *Applicant document* object type are not used in standard tasks. Instead, they are called up directly from the program when the optical archive is accessed. For this reason, this object type and its methods are not described here.

ApplicantActivity: Applicant Activity

An applicant activity represents an administrative stage through which an applicant passes during the selection procedure. An applicant activity is characterized by the following features:

ApplicantActivity: Applicant Activity

- **Activity type**

The applicant activity type determines the activity that must be performed for an applicant (for example, *Mail confirmation of receipt*, *Mail invitation to interview*, *Appointment for interview*).

- **Activity status**

The status of the applicant activity indicates whether the activity has already been performed for the applicant (*completed*) or has yet to be performed (*planned*).

- **Performance date**

If applicant activities are planned, the performance date is the date by or on which they must be performed. If applicant activities are completed, this is the date on which they were performed.

- **Person responsible for activity**

This person is responsible for ensuring that the applicant activity is performed on or by the date specified.

Applicant activities enable you to perform the following functions for an applicant:

- Managing appointments (for example, *job interview*), with the option of entering additional information (for example, building, room) or documents (for example, log).
- Processing applicant correspondence by including standard letters.
- Transferring applicant data to *Personnel Administration*.

The business application object of the applicant activity is represented by the *Applicant activity* (ApplicantActivity) object type. This object is uniquely identified by the applicant number, the date of receipt of the application, the sequential number of the application, and the unique identification number of the applicant activity.

Attributes

When the workflow is run, the following fields are available as attributes:

- *Performance date*
- *Performance time*
- *Building number*
- *Room number*
- *Status planned/completed*

The following objects are also available as attributes:

- *Application* (aggregate) to which the applicant activity refers.

Events

The *Applicant activity* object type can have the following statuses:

- *Applicant activity created*
- *Applicant activity completed*
- *Applicant activity deleted*

Display Applicant Activity

- *System message generated* (if an error occurs when the applicant activity is created in the background).

This event is used in workflow templates 00400019 and 00400020.

Each change of status can be included in the Workflow as an event.

Object Methods

The following methods can be performed for the *Applicant activity* object type:

- **Display (ApplicantActivity.Display)**
- **Maintain (ApplicantActivity.Maintain)**



Applicant activities are created using the *Create applicant activity without dialog* (Application.ActivityCreateAutom) and *Create applicant activity with dialog* (Application.ActivityCreate) methods for the [application \[Page 21\]](#) object type.

Display (ApplicantActivity.Display)

The *Display* object method corresponds to the [Display Applicant Activity \[Page 26\]](#) transaction in the application.

Maintain (ApplicantActivity.Maintain)

The *Maintain* object method corresponds to the [Change Applicant Activity \[Page 27\]](#) transaction in the application.



In the standard scenarios in the Recruitment System, the *Maintain* object method is used, for example, to change the performance date for the *Appointment for interview* applicant activity if the applicant is unable to attend on the date specified and a new appointment is arranged.

The *Maintain* object method is used in the following standard tasks:

- TS00407998 *Coordinate interview*
- TS00407892 *Print and mail invitation*
- TS00008280 *Edit contract*
- TS00407894 *Mail contract*
- TS00407896 *Create log for interview*

Display Applicant Activity

1. Choose *Applicant activity* → *Display*.

Change Applicant Activity

The *Display Applicant Activities* screen appears.

2. In the *Applicant number* field, enter the applicant number that you require.
3. Choose *Execute*.

Does the applicant have more than one application?

If **yes**, select the required application and choose *Choose*.

If **no**, proceed directly to step 4.

4. The system now displays a second *Display Applicant Activities* screen, listing all of the planned and completed activities for a selected applicant.

Select the required activity.

5. Choose *Display activity*.

The system displays the *Display Planned Activity* or *Display Completed Activity* dialog box.

Result

The required applicant activity is displayed.

Display Notes on Applicant Activity

6. Choose *Display notes*:

This takes you to the *Display Notes on Activity: <activity> language EN* screen.

Result

The system displays the notes on the required applicant activity.

Change Applicant Activity

Prerequisites

If you do not want to create or change notes on the activity, you can skip steps 8 to 10.

Procedure

1. Choose *Applicant activity* → *Maintain*.

The *Maintain Applicant Activities* screen appears.

2. In the *Applicant number* field, enter the applicant number that you require.
3. Choose *Execute*.
4. Does the applicant have more than one application?
If **yes**, select the required application and choose *Choose*.

VacancyAssignment: Applicant Vacancy Assignment

If **no**, the system displays a second *Maintain Applicant Activities* screen listing all planned and completed activities for a selected applicant.

5. Select the activity that you require, and choose *Change activity*.

This takes you to the Change Planned Activity or Change Completed Activity dialog box.

6. Maintain the fields on this screen.

7. If necessary, select *Send mail*.

8. Choose *Maintain notes*.

This takes you to the Change Notes for Activity: <activity> language EN screen.

9. Maintain your notes, and then save.

10. Choose *Back*.

This takes you back to the Change Planned Activity or Change Completed Activity dialog box.

11. Choose *Transfer*.

The *Maintain Applicant Activities* screen is displayed again.

12. Save.

Result

The applicant activity has been changed. You have also sent a mail about changing the activity, if you selected this parameter.

VacancyAssignment: Applicant Vacancy Assignment

Infotype 4002 *Vacancy Assignment* is used to record the vacancies to which an applicant is assigned. Assigning an applicant to a vacancy means that he or she takes part in the selection procedure for this vacancy. An applicant's status in the selection procedure for a vacancy is indicated by his or her vacancy assignment status for the vacancy in question.

One applicant can be assigned to the selection procedures of more than one vacancy at the same time.

If this is the case, each vacancy assignment must be assigned a different priority. There must also be a vacancy assignment status for each of the applicant's vacancy assignments.

The business application object of the vacancy assignment contains the vacancy, the vacancy assignment status, and the reason for this status. This is represented by the *Vacancy assignment* object type, which is uniquely identified by the applicant number and the vacancy assignment priority.

Attributes

When the workflow is run, the following fields are available as attributes:

- *VacancyID* (position key)

VacancyAssignment: Applicant Vacancy Assignment

- *Status of applicant's vacancy assignment*
- *Reason for status*
- *Vacancy description*

The following objects are also available as attributes:

- *Applicant* (aggregate of object)

Events

The *Vacancy assignment* object type can have the following statuses:

- *Vacancy assignment created*

Each change of status can be included in the Workflow as an event.

Each change to an applicant's vacancy assignment status can also be included in the Workflow as an event (for example, applicant put on hold for vacancy, contract offered, applicant rejected, etc.)

Object Methods

The following methods can be used for the *Vacancy assignment* object type:

- **Change Vacancy Assignment Status (VacancyAssignment.ChangeStatus)**
- **Selection Decision (VacancyAssignment.Decide)**

Change Vacancy Assignment Status (VacancyAssignment.ChangeStatus)

The vacancy assignment status represents the applicant's status in the selection procedure for a vacancy. The system displays all of an applicant's vacancy assignments, which can then be edited.

The *Change vacancy assignment status* object method is not performed online.

The *Change vacancy assignment status* object method changes an applicant's vacancy assignment status using report RPAPUT00.

The *Change vacancy assignment status* object method has the following parameters:

- *Status* = New vacancy assignment status
- *StatusReason* = Status reason for vacancy assignment status
- *IDVacancy* = Identification number of vacancy

You must specify a status.



In the standard scenarios in the Recruitment System, the *Change vacancy assignment status* object method can be used by the personnel officer, for example, to reject an applicant for a vacancy.

The *Change vacancy assignment status* object method is used in the following standard tasks:

- TS00008288 *Reject applicant for vacancy*

Preparation and Customizing (PA-RC)

Selection Decision (VacancyAssignment.Decide)

The decision to select applicants is documented by a change in the vacancy assignment status. This method displays all of the applicants for a vacancy, enabling you to change their vacancy assignment status.

The *Selection decision* object method is performed online.

The *Selection decision* object method changes an applicant's vacancy assignment status using report RPAPUT01.

The *Selection decision* object method has the following parameters:

- *ApplicantNumber* = Applicant number
- *Advertisement* = Job advertisement
- *Vacancy* = Vacancy



In the standard scenarios in the Recruitment System, the *Selection decision* object method can be used by the line manager, for example, to create a list of applicants in order to decide which applicants to invite for an interview.

The *Selection decision* object method is used in the following standard tasks:

- TS00008249 *Decision regarding interview*
- TS00407890 *Decision to offer contract*

Preparation and Customizing (PA-RC)

Use

While the general customizing process contains steps for setting up the Workflow System, you may also need to modify the standard workflow templates provided for the Recruitment System.

Features

The following three cases may apply to customizing, depending on your requirements:

1. **SAP provides processes that are the same as those you require, or similar.**
Copy the standard workflow templates to create workflow tasks of your own, and modify them.
2. **SAP does not provide processes that are the same as those you require, or similar.**
Create your own workflow tasks using the existing standard tasks or your own customer tasks. You can use the existing object methods and events to create these customer tasks. Use the standard tasks as a template when creating new customer tasks.
3. **You need modules (objects, methods, events, roles) that are not provided with the standard system.**

Specific Customizing Using Examples

Create your own modules by modifying the standard objects.

This documentation describes how to perform points 1 and 2. This description of how to perform customizing is based on four areas that customers must adapt, as experience as shown.

If you require documentation on point 3, please see the appropriate section in the R/3 Library (technical description of workflow objects in [BC - SAP Business Workflow \[Ext.\]](#)).

See also:

[Specific Customizing Using Examples \[Page 31\]](#)

Specific Customizing Using Examples

This section describes how to modify the standard workflow templates and standard tasks using examples.

[Setting Parameters for Applicant Activities \[Page 31\]](#)

[Differentiating Between Letters for Applicant Correspondence \[Page 34\]](#)

[Adding New Tasks \[Page 37\]](#)

[Modifying Deadline Monitoring \[Page 39\]](#)



Do not change any standard objects. Instead, copy them to create customer objects and make your changes there. This ensures that your changes and additions are not overwritten by a release upgrade.

Parameter Settings for Applicant Activities

Use

The administrative steps carried out for an applicant, including correspondence, are represented by applicant activities.

Activities are created with model parameter settings in the standard workflow templates. These settings also include the names of standard texts used for these applicant activities. It is usually a good idea to use names other than those in the standard system for your standard applicant correspondence texts.



New Standard Text for Rejection Letter

Setting Parameters for Applicant Activities

In the standard system, the standard **REJECTION** text is used for a rejection letter sent to an applicant. However, you have created your own standard text for a rejection and want it to be used in the appropriate places in the workflow, instead of the standard **REJECTION** text. If you require information on the procedure for this example, see [New Standard Text for Rejection Letter \[Page 33\]](#).

See also:

[Setting Parameters for Applicant Activities \[Page 32\]](#)

Setting Parameters for Applicant Activities

Prerequisites

In the Recruitment System, you can record a text name for an applicant activity so that the system creates an applicant letter. Two object methods can be used to create applicant activities:

- **Application.ActivityCreateAutom:** create activity without dialog
- **Application.ActivityCreate:** create activity with dialog

They are assigned the appropriate values (such as the name of a standard text) during binding.

The applicant activity itself is created in a batch job.

If an error occurs, the applicant activity cannot be created in the batch job. The standard system response in this case is to process the applicant activity online, which gives you the opportunity to add missing data, or correct the error.

This is not possible when data is processed in batch.

For this reason, the above object methods are not used to create letters. Instead, the standard system contains two workflow templates in which errors are processed.

Letters are output in the following workflow templates:

- WS00400019 *Create letter*
- WS00400020 *Create + mail letter*

If an error occurs when an applicant activity is created in batch, these workflow templates create an appropriate event. The agent for the task receives a work item that can be used to create the applicant activity online.



Use these workflow templates whenever applicant activities and their accompanying letters are created.

Procedure

1. Find the workflow step you want to change in the standard workflow templates.

New Standard Text for Rejection Letter: Example

2. Copy the entire workflow template (see the technical documentation on [creating and maintaining tasks \[Ext.\]](#)).
A definition of the values used in a workflow step is included in the binding definition for the step in question.
3. Change the values in the binding definition according to your requirements.
4. Save and activate the new workflow template.
The workflow template you created is used instead of the workflow template supplied by SAP.
5. Deactivate event linkage for the SAP workflow template, and activate it for the new workflow template. To do so, switch to edit mode for the pertinent multistep tasks. If you require further information, see [Creating a Triggering Event \[Ext.\]](#).

See also:

[New Standard Text for Rejection Letter \[Page 33\]](#)

New Standard Text for Rejection Letter: Example

Prerequisites

The rejection letter is created in workflow step 131 of template WS00200038. This workflow step transfers the necessary data (including the name of the standard text) to sub-workflow WS00400019.

In the standard system, the standard letter for a rejection specified in the binding definition for workflow step 131 is the **REJECTION** text.

Procedure

1. Copy workflow template WS00200038 to create your own workflow task.
2. In this workflow task, change the binding values according to your requirements. Replace the standard **REJECTION** text with the name of your own standard text for the rejection letter.
3. Save and activate the workflow task.
4. Deactivate the event linkage for workflow template WS00200038, and activate it for your new workflow task.

Result

Workflow step 131 no longer assigns the standard **REJECTION** text to sub-workflow WS00400019 for creating letters. Instead, it assigns the text you recorded.

Differentiating Between Applicant Correspondence

Differentiating Between Applicant Correspondence

Use

If you want to make greater distinctions between the standard letters created for applicant correspondence according to certain criteria, you must modify the existing workflow template.



Differentiating Between Rejection Letters

You want to send different letters to applicants depending on the reasons for which they are rejected.

For example, if applicants are rejected due to formal errors in their applications, they are only sent a relatively short letter. However, if they are rejected because they are overqualified, the reasons for the rejection are explained in detail. You have two different standard rejection letters for these purposes, and want them to be used at the appropriate points in the workflow. If you require information on the procedure for this example, see [Differentiating Between Rejection Letters \[Page 35\]](#).

See also:

[Differentiating Between Letters for Applicant Correspondence \[Page 34\]](#)

Differentiating Between Applicant Correspondence

Prerequisites

Letters are normally printed using workflow templates WS00400019 and WS00400020.

To keep a clear overview, it is not a good idea to make distinctions between letters within these templates, but in a separate workflow task instead.

Insert this new workflow task as a sub-workflow in the appropriate main workflow before printing letters (that is, before workflow templates WS00400019 or WS00400020 are called up).

Procedure

1. Create a new workflow task to determine the rejection letters. (For further information on creating new workflow tasks, see the technical documentation on [creating and maintaining tasks \[Ext.\]](#).)
2. Define the container for this new multistep task. Check that it includes all of the elements that you require. You can add missing elements to the container.
3. Define a multiple condition as the first step in the task. At this point, the system queries the element in the container, which determines the distinction between different letters.

Depending on the result of this comparison operation, the system determines the name of a standard text and transfers it to a subsequent workflow task (such as WS00400019).

Differentiating Between Rejection Letters: Example

4. To create your own workflow task, copy the main workflow in which the new sub-workflow must be inserted.
5. Insert the task you have just created in the copy of the main workflow.

See also:

[Differentiating Between Rejection Letters \[Page 35\]](#)

Differentiating Between Rejection Letters: Example

Prerequisites

Make the required distinctions between rejection letters on the basis of the status reason, which can be set when the applicant is rejected (overall status changed to 4).

The following rejection letters must be printed on the basis of the various status reasons:

Status reason	Text ID
01 formal errors	REJECT1
02 overqualified	REJECT2

Procedure

1. Create a new workflow task to determine the rejection letter. You must add the following two elements to the container for this new workflow task:
 - The *Applicant* object type as an import element (which contains the status reason as an attribute)
 - The *Standard text* element from the Data Dictionary as an export element
2. When you create a new container element, the system asks you to specify whether the new element must be linked to a Data Dictionary field or an object type. The import element for the applicant is linked to the *Applicant* object type, and the export element for the rejection letter to the *PC3AC-IDTXT* field in the Data Dictionary.
3. Access the graphical or alphanumeric editor.
4. Insert a multiple condition (CASE) as the first step in the workflow task.

Create this multiple condition with the following data:

Step name	Text of your choice
<i>Basis of comparison</i>	&APPLICANT.STATUSREASON& as a workflow container element (attribute of the <i>Applicant</i> object type, which can be selected using F4 help)
<i>Values to be compared</i>	Possible status reasons (which can be selected using F4 constants)
<i>Event description</i>	Description of status reasons (as a description of the various branches of the condition)

Differentiating Between Rejection Letters: Example

<i>Other values</i>	Description of the branch for all other values
---------------------	--

5. Use the green arrow to exit the screen.

Result:

The decision operation has a branch for each value to be compared. The steps and events that follow on from the possible values to be compared for the decision have not yet been defined.

6. Select the step that you now want to define. Define this step as a container operation (with a double click, or by choosing Edit - Create).
7. Create the container operation using the following data:

<i>Texts for step and event</i>	Text of your choice
<i>Result element</i>	Choose F4 to select the standard text element (IDTEXT) from the container.
<i>Printing</i>	Name of the standard text to be assigned (for example, REJECT1).

All other fields are left blank.

Result:

You have defined the workflow step that follows on from the result of the decision operation, and the subsequent event.

8. Make definitions for all branches of the decision operation, and assign the appropriate values for the standard texts.
9. Save and activate this new workflow task.
10. Copy the main workflow, applicant processing (WS00200038), to create your own workflow task.
11. Between the *Rejected* event and the *Create + mail rejection letter* workflow step, insert a new *Activity* workflow step.
12. Assign this new step the *Determine rejection letter* workflow task, which you created.
13. You must now ensure that the *Create + mail rejection letter* workflow step receives the standard text that was determined in the *Determine rejection letter* sub-workflow.
- To do this, you must add the *Standard text* element to the container for the main workflow. Then replace the **REJECTION** constant in the binding of the *Create + mail rejection letter* workflow step with the *Standard text* container element.
14. Save and activate the new workflow task.
15. Finally, you must ensure that the new workflow task is used to record a new application instead of workflow template WS00200038.

To do this, activate event linkage for the new workflow task, and deactivate it for workflow template WS00200038 (see technical documentation: [BC - SAP Business Workflow \[Ext.\]](#)).

Adding New Tasks

Use

The tasks within the workflow represent individual methods as required for business purposes. The standard system contains numerous tasks that enable you to perform all of the business procedures for Recruitment contained in the SAP Reference Model.

If you require new tasks in addition to those provided in the standard system, you can create your own customer tasks. To do this, copy the standard tasks and change the copies according to your requirements.



New Applicant Activity: Appointment for Aptitude Test

As well as job interviews, your company also conducts aptitude tests. You want to use applicant activities to record test appointments for specific applicants.

To do this, you have created new activity type 999 *Test appointment*. This applicant activity must be created as part of the workflow scenario. If you require information on the procedure for this example, see [New Applicant Activity: Appointment for Aptitude Test \[Page 38\]](#).

See also:

[Adding New Tasks \[Page 37\]](#)

Adding New Tasks

Procedure

1. Find a similar standard task from among those provided with the standard system. Make sure you find a copy template that uses the same method as the one you want to use in the new task. This ensures that the task container is already defined according to your requirements.
2. Copy this standard task to create your own customer task (see the technical documentation on [creating and maintaining tasks \[Ext.\]](#)).
3. Change the following according to your requirements:
 - Work item text
 - Description of task (long text)
 - Parameter settings for binding
4. The binding enables you to enter fixed data or specify variables, which are then filled with values from the container when the workflow is run.

In this case, the following principle applies:

New Appl. Act: Appointment for Apt. Test: Example

- Data that determines the business contents of the task must be entered using constants.
This ensures that the information described in the text elements of a task is also contained in the data definition. In this way, you prevent the system from using the task to create an activity type different from that described in the text, due to values being retrieved from the workflow container.
 - The binding also contains elements that are only fixed when the workflow is run. When the task is defined, these elements are stored as variables. They are filled during the workflow with values from the container.
5. Save the new task.

Result

You can now use the new task in the appropriate places in the workflow steps.

See also:

[New Applicant Activity: Appointment for Aptitude Test \[Page 38\]](#)

New Appl. Act: Appointment for Apt. Test: Example**Procedure**

1. You have created new activity type 999 *Test appointment*.
You cannot use any of the standard tasks to create this applicant activity in the workflow, and must therefore create a suitable customer task.
2. To create an applicant activity, you can use the **ActivityCreateAutom** (create activity without dialog) or **ActivityCreate** (create activity with dialog) object method. If you want to specify the date of the appointment when creating the applicant activity, select the **ActivityCreate** object method.
3. Find a similar standard task that uses the same method (for example, standard task TS00008244 *Create interview appointment*), and use it as a copy template.
4. Copy this standard task to create your own customer task.
5. Change the ID and name of the copy of the task. Then enter a work item text and task description that clearly describe the content of the new task for the user.
6. Now enter constants or variables in the task binding. Please take note of the principle explained in [Add New Tasks \[Page 37\]](#).

In this case, enter the following specifications for the elements in the binding:

<i>IndicCompPlann</i>	S
<i>Action</i>	999
<i>StandardText</i>	
<i>RecurringTask</i>	

Modification of Deadline Monitoring

<i>PerformanceDate</i>	
<i>Time</i>	
<i>ObjectTypeVacancy</i>	&OBJECTTYPEVACANCY&
<i>Vacancy</i>	&VACANCY . VACANCY&
<i>ReferenceActionID</i>	

7. Save the new task.

Result

You have defined the new task for creating activity type 999 *Test appointment*. The binding definition, work item text, and long text provide a clear description of the task's business content.

Modification of Deadline Monitoring

It is possible to set a deadline for every step in a workflow definition. You can set a deadline for the following:

- **Requested start**

The time specified here is the earliest possible start for processing a work item, that is, the earliest time at which the work item may be placed in an agent's inbox.

If you specify a requested start, the work item is first created with the *waiting* status (for example, when a particular event begins), which changes to *ready* when the requested start is reached.

Dialog work items initially appear with this status in the selected agents' integrated inboxes. Other work items with this status are carried out by the system.

- **Latest start**

The time specified here indicates the latest time at which work item processing may begin.

- **Requested end**

A work item is regarded as completed when its status changes to *completed*.

The time specified here indicates the latest time at which work item processing may end.

- **Latest end**

A work item is regarded as completed when its status changes to *completed*.

The time specified here indicates the latest time at which work item processing may end.

Deadlines are commonly set at the following stages in the standard workflow templates:

- Resubmission if an applicant does not have an open vacancy assignment (applicant selection)
- Length of time that applicants can remain on hold in the applicant pool
- Punctual return of contract

Modifying Deadline Monitoring

- Recording of interview log

If one of the deadlines is exceeded, the system can respond in the following two ways:

- The person who has exceeded the deadline is sent a notification in the form of a work item. The text for this notification is standard.
- A customer-defined event is triggered. This event can be sent to a task. This enables you to model process flows as a reaction to a deadline violation.



Modifying Applicants' Length of Time in Applicant Pool

The length of time applicants with the *on hold* status may remain in the applicant pool is set to 3 minutes in the standard system. You must change this value to suit your requirements. If you require information on the procedure for this example, see [Modifying Applicants' Length of Time in Applicant Pool \[Page 41\]](#).

See also:

[Modifying Deadline Monitoring \[Page 40\]](#)

Modifying Deadline Monitoring

Setting Deadlines

Deadlines are created or changed directly in the individual steps within a workflow template.

1. Find the template in which you want to change the deadline.
2. Edit the workflow template in the graphical or alphanumeric editor. Select the step for which the deadline must be changed.

Under *Deadlines*, the system lists the above options for setting a deadline (*requested start*, *latest start*, *requested end*, *latest end*).

3. Enter the required deadlines under Offset.
4. If you want the system to create an event if the deadline is exceeded so that other process flows are triggered, you must specify the required event.

Responsibility when Deadlines are Exceeded

5. If you only want the system to send a notification in the form of a work item when a deadline is exceeded, maintain the person responsible under *Responsibility*.

See also:

[Modifying Applicants' Length of Time in Applicant Pool \[Page 41\]](#)

Modifying Applicants' Length of Time in Appl. Pool

1. Copy the WS00200034 *Management of applicants on hold* workflow template. If you require further information, see [Creating and Maintaining Tasks \[Ext.\]](#).
2. Edit this workflow template using the Workflow Builder. Find step 000012 *Reject applicant when deadline reached*.
3. Maintain the workflow step. The *Requested start* tab page contains an entry of 3 minutes. Change this value according to your requirements.
4. Save and activate the workflow template.
5. Deactivate event linkage for the WS00200034 workflow template, and activate it for the new workflow template. If you require further information, see [Creating a Triggering Event \[Ext.\]](#).

Using and Linking to Application Functions (PA-RC)

[Start/End of Workflow \[Page 41\]](#)

[Controlling the Workflow Using the Applicant Status \[Page 41\]](#)

[Controlling the Workflow Using the Vacancy Assignment \[Page 42\]](#)

[Using Applicant Activities in Workflow \[Page 43\]](#)

Start/End of Workflow

The workflow is started when a new [application \[Page 24\]](#) is created.

The workflow is only complete when the applicant is either

- Rejected, or
- Hired

Controlling the Workflow Using Applicant Status

The most important criterion for controlling the workflow is the applicant's overall status.

All changes to an applicant's overall status are available in the system as events, which can be included in the workflow from the application (by performing work items or transactions outside of the workflow).

Controlling the Workflow Using Vacancy Assignment

The following events from the application are currently used in the workflow provided with the standard system:

- **Applicant rejected**

The workflow can include this event from the application at any time. When this event is triggered, all of the sub-workflows that are active for the applicant (such as *Management of Job Interviews*) are canceled. At the same time, the system creates a rejection letter for the applicant. The workflow ends on completion of the *Rejection letter* applicant activity (that is, when a rejection letter is mailed).

- **Applicant in process**

This event ends the workflow used to monitor applicants on hold. If a vacancy assignment takes place when the applicant status is changed to in process, the person responsible for the position must decide whether the applicant should be invited to a job interview, or rejected. If the status is all that changes, the personnel officer must decide on a further procedure for the applicant.

Changes to the applicant's overall status are made as tasks parallel to the workflow. The following status changes are currently available as tasks in the standard system:

- **Applicant offered contract**

If the person responsible for the position decides to offer the applicant an employment contract as a result of the interview, the personnel officer must set the applicant's overall status to Contract offered. He or she can also change the vacancy assignment status of all other positions assigned to the applicant.

- **Applicant to be hired**

When the personnel officer confirms that the employment contract has been signed by the applicant and returned, he or she must set the applicant's overall status to To be hired. The following options are also available:

- Changing the vacancy assignment status of all other positions assigned to the applicant.
- Changing the vacancy assignment status of all other applicants assigned to the position.

See also:

[Preparing to Hire an Applicant \[Page 15\]](#)

Controlling the Workflow Using Vacancy Assignment

The most important criterion for controlling the selection procedure for an applicant is the assignment of that applicant to one or more vacancies.

Vacancy assignments can be made either via the workflow (receipt and resubmission), or via the application itself (for example, initial entry of basic data, search for suitable applicants using qualifications profiles).

Use of Applicant Activities in Workflow

The selection procedure is triggered for an assigned vacancy. If several vacancies are assigned, the system starts the selection procedure for the vacancy with the highest priority.

As the first step in the selection procedure, the person responsible for the position checks the applicant.

If the applicant is rejected for the vacancy as a result of this check or later on in the procedure (job interview, contract offer, etc.), the system determines the vacancy assignment with the next highest priority. The next selection procedure is triggered for this vacancy. Therefore, the processing of applicants for vacancies assigned to them is successive, rather than parallel.

If the applicant has no other valid vacancy assignments (that is, ones that have not been rejected), he or she is transferred to receipt and resubmission after rejection.

An applicant can be rejected for a vacancy via the workflow, or via the application at any point in the selection procedure. Workflow information on the applicant's rejection is effected via an event.

Use of Applicant Activities in Workflow

You can use all applicant activity types in your customer workflow. Applicant activities can be created online or in the background. The workflow is informed that a certain applicant activity has been completed (either in the application or as a work item) by means of event linkage.



Creating and Mailing Applicant Letters

An applicant activity of the appropriate type is created in background processing for all applicant letters (for example, confirmation of receipt, letter of rejection). The parameters (standard text, completion date, etc.) are recorded directly in the workflow definition via binding.

When the applicant activity is created, the system creates a wait step that ends when the applicant activity is completed. Letters are printed and the applicant activity completed independently of the workflow, either in the application dialog or using the recurring tasks function.

Letters inviting applicants to interviews can be printed, and the applicant activity ended, using a work item. This work item is placed in the personnel officer's inbox. If a letter is printed and the applicant activity ended using the recurring tasks function, the work item is automatically removed from the inbox.

In addition to processing applicant correspondence, workflow also uses applicant activities for the following:

- **Creating and mailing the draft contract**
- **Setting the appointment for a job interview**
- **Hiring date**
- **Scheduling data transfers**

Workflow Scenarios in Pers. Administration (PA-PA)



When setting up the system, please note that you do not need to create and set parameters for applicant activities using feature PACTV.

For more information on applicant activities, see the component documentation on [Recruitment \[Ext.\]](#).

Workflow Scenarios in Pers. Administration (PA-PA)

This section describes how SAP Business Workflow is used in *Personnel Administration*.

[Follow-Up Activities for a New Hire \(PA-PA\) \[Page 44\]](#)

[ESS - Follow-Up Activities for a New Hire \(PA-PA\) \[Page 46\]](#)

[Delimiting an ESS User \(PA-PA\) \[Page 48\]](#)

[Approving an Individual Basic Pay Increase \(PA-PA\) \[Page 49\]](#)

Follow-Up Activities For Hiring (PA-PA)

Purpose

The *Follow-Up Activities for Hiring* workflow represents the following process:

After a new employee has been hired, a series of administrative and technical steps must be taken before work commences so that he or she is integrated in the company's organizational structure and infrastructure. Different employees are responsible for different steps.

Process Flow

Such steps could, for example, include the following activities and responsibilities.

The departmental secretary determines the employee's building number and room number, and how the work center should be equipped. The technical department assigns the new employee a telephone number, the PC service group installs the necessary hardware, and the security department issues the ID card.

A work commencement letter is created for the employee automatically. As well as the personal data of the employee that is required, the letter includes the data entered during workflow such as the ID card number, building number, and room number. The work item for printing the letter is sent to the inbox of the employee responsible.

Graphic: [Workflow "Follow-Up Activities For Hiring" \[Page 46\]](#)

Advantages, Improvements, Simplifications

The *Follow-Up Activities for Hiring* workflow facilitates efficient company procedures. It supports and accelerates the process described above in accordance with an enterprise's specific requirements by linking the necessary steps, and sending them automatically to the agent.

Follow-up activities for hiring can be generally applicable and flexible. All of the employees concerned are provided at their work centers with the information they require, taking account of who needs to know what, and when they need to know it. This avoids the duplication of work, and enables you to monitor and cope with time-crunch situations. The coordination of individual steps can be optimized so that, for example, the PC service group is not requested to enter a PC equipment number until the departmental secretary has determined how the work center should be equipped. It is possible to ensure that all new employees are properly informed about their coming work activities.

IDES

The *Follow-Up Activities for Hiring* workflow scenario is included in IDES:

Help → R/3 library → Demo System (IDES) → Cross Application → Follow-Up Activities for Hiring.

Using and Linking to Application Functions (PA-PA)

Features

The *Follow-Up Activities for Hiring* process can be regarded as an example of how workflow technology reflects and facilitates core HR procedures.

It includes employee events that represent important core procedures, such as when an employee is *hired*, or *leaves* the company, or is *reassigned* within an organization.

It also includes the most important infotypes as business objects. As a result, all tasks concerned with the maintenance of employee data can be used within such workflows.

Activities

The *Hiring* personnel action (event: *Employee.hired*) triggers workflow automatically. A work item is sent to the inbox of the employee responsible for the administrative or technical task in question. The work item requests the employee to perform the activity.

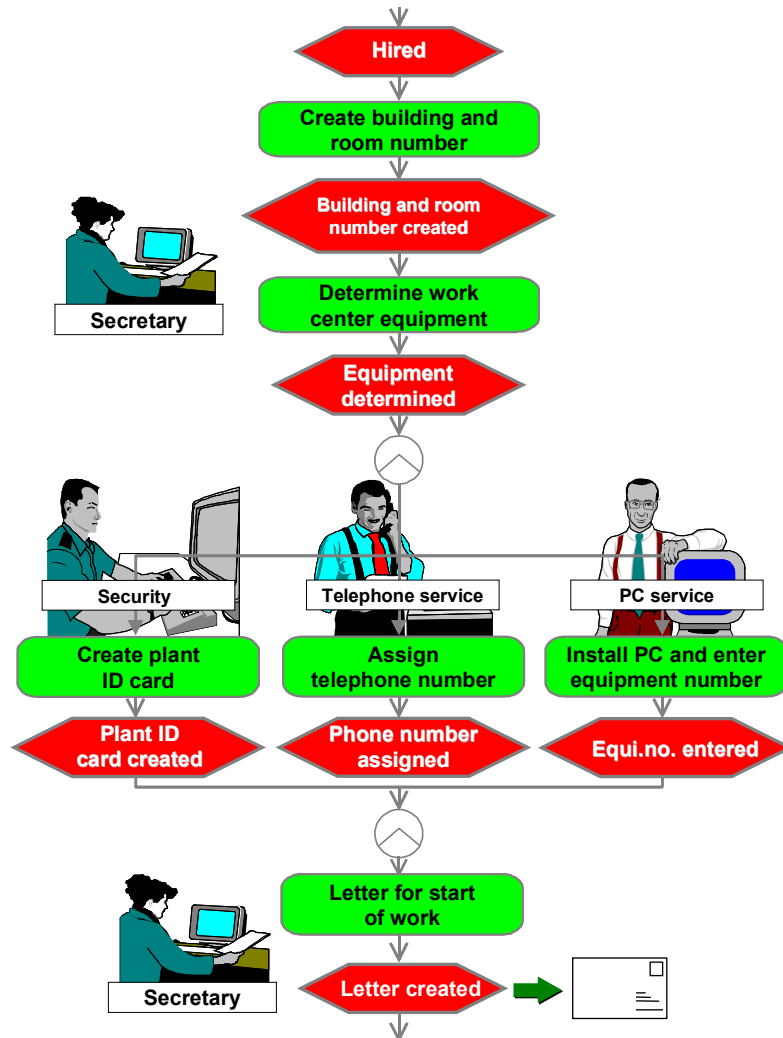
The fact that a task has been executed outside the system itself (such as when an ID card is issued) is documented by maintaining the appropriate infotype record (in this case, by entering the ID card number in infotype 0032 *Internal Control*).

The work item is removed from the inbox once the infotype has been maintained and processing confirmed.

The work commencement letter is created using a SAPscript document. All data on the *employee* business object can be included in this standard document, including data added to the new employee's master data record during workflow. The work commencement letter is not written to the printer automatically. Instead, it is sent as a concluding work item to the inbox of the person responsible.

Workflow "Follow-Up Activities for Hiring"

Workflow "Follow-Up Activities for Hiring"



ESS - Follow-Up Activities For a New Hire (PA-PA)

Purpose

The *ESS - Follow-Up Activities for a New Hire* workflow represents the following process:

Using and Linking to Application Functions (PA-PA)

After a new employee has been hired, a series of administrative and technical steps must be taken before work commences so that he or she is integrated in the company's organizational structure and infrastructure.

Implementing **Employee Self-Service (ESS)** enables you to include your new employees directly in this process because they can then maintain some of their HR data themselves. The task of entering HR data for new hires is, therefore, shared between the HR Department and the new employees themselves.

Process Flow

The following activities and responsibilities, for example, are possible:

The system administration department at your enterprise creates an ESS user for the new hire. This provides the employee with the system authorizations required for using the services included in ESS.

At the same time, the new employee is required to supplement his or her HR data in the [New Hire Data in the Internet \[Ext.\]](#) service. Once the employee has entered his or her missing HR data, the appropriate personnel administrator checks that the data entered by the employee is both complete and plausible.

Advantages, Improvements, Simplifications

The *ESS - Follow-Up Activities for a New Hire* workflow facilitates efficient company procedures. It supports and accelerates the process described above in accordance with an enterprise's specific requirements by linking the necessary steps, and sending them automatically to the agent responsible.

It ensures that an ESS user is assigned to each new hire, and that the employee has the system authorizations required for supplementing his or her own HR data immediately after the hiring has taken place. As a result, the employee assumes responsibility for his or her own data just as soon as hiring takes place, and relieves the HR Department of the burden of expensive, time-consuming activities.

The appropriate personnel administrator is responsible for ensuring that data is correct.

Using and Linking to Application Functions (PA-PA)

Features

The *ESS - Follow-Up Activities for a New Hire* process can be regarded as an example of how workflow technology reflects and facilitates core HR procedures.

It includes employee events that represent important core procedures, such as when an employee is *hired*, or *leaves* the company, or is *reassigned* within an organization.

It also includes the most important infotypes as business objects. As a result, all tasks concerned with the maintenance of employee data can be used within such workflows.

By integrating steps that are processed in the web inbox, workflow options are considerably enhanced. These processes can include not only the personnel administrators, but also all of the employees who have an ESS user.

Delimiting an ESS User (PA-PA)

Activities

The *Hiring* personnel action (events: *Employee.hired*, *Employee.rehired*) triggers workflow automatically.

A work item is sent to the inbox of the system administrator responsible for assigning ESS users. It requests that the new employee be assigned an ESS user. By executing the work item, the system administrator accesses the dialog used to create ESS users. The work item is removed from the inbox once the ESS user has been created, and processing of the work item has been confirmed.

The new hire logs onto ESS with his or her new user. The employee finds a work item in his or her web inbox. It requests that the employee supplements his or her HR data in the [New Hire Data in the Internet \[Ext.\]](#) service. Once the data has been entered, the employee returns to the web inbox and confirms that he or she has supplemented the hire data.

The personnel administrator responsible then receives a work item in his or her inbox. It requests him or her to check the data that has been entered. The personnel administrator now has two options:

- He or she confirms that the data is complete and plausible, which concludes the workflow.
- He or she requests that the data be corrected. The new hire receives a work item in his or her web inbox stating that the data needs to be corrected. Once the employee has corrected his or her HR data, the personnel administrator confirms that the data is complete and plausible, which concludes the workflow.

Delimiting an ESS User (PA-PA)

Purpose

The *Delimiting an ESS User* workflow represents the following process:

Once an employee has left the enterprise, his or her user for **Employee Self-Service (ESS)** must be delimited.

Process Flow

If an employee leaves your enterprise, the system administration department delimits the ESS user of the employee who has left. As of the leaving date, the employee is relieved of the system authorizations required for using the services included in ESS.

Advantages, Improvements, Simplifications

The *Delimiting an ESS User* workflow ensures that every employee who leaves the enterprise is immediately relieved of his or her ESS user. This prevents former employees from misusing ESS.

Using and Linking to Application Functions (PA-PA)

Features

The *Delimiting an ESS User* process can be regarded as an example of how workflow technology reflects and facilitates core HR procedures.

It includes employee events that represent important core procedures, such as when an employee is *hired*, or *leaves* the company, or is *reassigned* within an organization.

It also includes the most important infotypes as business objects. As a result, all tasks concerned with the maintenance of employee data can be used within such workflows.

Activities

The *Leaving* personnel action (events: *Employee.retired*, *Employee.company.left*) triggers workflow automatically.

A work item is sent to the inbox of the system administrator responsible for delimiting ESS users. It requests that the ESS user of the employee leaving the enterprise be delimited on the leaving date. By executing the work item, the system administrator accesses the dialog used to delimit ESS users. The work item is removed from the inbox once the ESS user has been delimited, and processing of the work item has been confirmed.

Approving an Individual BP Increase (PA-PA) (Obs.)



The *Approving an Individual Basic Pay Increase* workflow is obsolete. Therefore, it is no longer supported by SAP as of Release 4.5A.

Purpose

Processing data on specific employees is a sensitive subject within the R/3 System. For this reason, such data is usually processed centrally by personnel administrators with a special authorization to do so. They collect information on the employee that pertains to master data (such as personal data and payroll data) and working time (such as attendances, absences, and substitutions), and then enter it in the system.

To a certain extent, it is possible for the employees themselves to enter certain types of data in the system. This helps reduce the general administrative burden within the Human Resource department. Data relevant to payroll and future working time, however, must be approved. The employee's superior usually approves such data. Irrespective of this, the data often needs to be checked for plausibility and consistency by an employee with the authority to do so, such as an administrator within the HR department.

The *Approving an Individual Basic Pay Increase* workflow supports and accelerates the procedure described above in accordance with the specific requirements of your enterprise. It does this by linking the necessary approval and verification tasks, and then forwarding them automatically to the appropriate agent.

Using and Linking to App. Functions (PA-PA) (Obs.)

Process Flow

To reward an employee for commitment or performance by increasing his or her basic pay at a time other than the annual salary review, the employee's immediate superior can use the *Approving an Individual Basic Pay Increase* workflow.

The employee's immediate superior can use this workflow to make a request for an individual basic pay increase. The request is forwarded automatically to all of the persons who are responsible for approving pay increases for this organizational unit. The number of approval levels is unrestricted, which ensures maximum flexibility when you determine the number best suited to your enterprise structure. You can even include the works council in the approval procedure.

The persons involved in the approval procedure can choose between the following courses of action for the request:

- Approve
- Reject
- Postpone

If the request is rejected, the person responsible for the request can either withdraw or change it. If the request passes through all of the approval levels successfully, the increased basic pay record is recorded automatically in the system. Furthermore, the system creates a letter for the employee, informing him or her of the fact that an increase in basic pay has been approved and implemented.

Using and Linking to App. Functions (PA-PA) (Obs.)



The *Approving an Individual Basic Pay Increase* workflow is obsolete. Therefore, it is no longer supported by SAP as of Release 4.5A.

Features

The *Approving an Individual Basic Pay Increase* process can be regarded as an example of how workflow technology reflects and facilitates core HR procedures.

Activities

The following example illustrates how workflow is performed in a single-level approval procedure.

The following persons are all involved in the workflow at various stages:

- The employee's disciplinary superior as the instance responsible for making the request
- The head of the organizational unit as the instance responsible for approving the request
- The personnel administrator as the instance responsible for checking the request

Using and Linking to App. Functions (PA-PA) (Obs.)

Entering a Request for an Individual Increase in an Employee's Basic Pay

As the person responsible for making the request, the employee's superior retrieves a request form from his or her local PC. The superior enters the name of the employee whose pay should be increased in this form. During the remainder of the procedure, the form accesses the R/3 System directly. Using the employee's name, the employee's current basic pay is taken from infotype 0008 *Basic Pay* and presented to the superior on the form for further processing. To increase the current basic pay, the superior can change existing wage types or create new ones. The superior can then write a note to justify the request.

By choosing *Send*, the request is forwarded to the head of the organizational unit. Sending the request triggers workflow automatically.

Rejecting, Postponing, Approving a Request for an Individual Increase in Basic Pay

As the instance responsible for issuing approval, the head of the organizational unit receives a work item in his or her inbox. The work item requests him or her to check the request.

By executing the work item, the request form is started on the local PC. At the same time, the current data in the request is read to the form. Using tab pages, the head of the organizational unit can compare the employee's current pay with the increased payments that have been requested.

The head of the organizational unit can choose between the following courses of action for the request:

- Reject
- Postpone
- Approve

If the request is rejected, a note can be used to justify the decision, or request a correction for the request.

The Request for an Individual Increase in Basic Pay was Rejected

If the head of the organizational unit does not approve the request, the employee's superior receives a work item in his or her inbox. The work item informs the superior of the fact that the request was rejected. By executing the work item, the superior can display the request form again. Two options are now available:

- **Cancel** the request for an individual increase in basic pay:
The superior cancels the request, which concludes the workflow.
- **Change** the request for an individual increase in basic pay:
The superior changes the request according to the wishes of the head of the organizational unit. Once again, the superior can use a note to comment on the correction.

By sending the request, the approval procedure is restarted.

The Request for an Individual Increase in Basic Pay was Postponed

If the head of the organizational unit postpones the request, the superior is informed by mail and asked to resubmit the request on a specific date.

The superior receives a work item in his or her inbox on this date. By executing the work item, the previous request form is displayed with all notes. The superior decides whether to change the request or not, and then sends it again to the head of the organizational unit.

Process Leave Request (PA-TIM-REC)

By sending the request, the approval procedure is restarted.

The Request for an Individual Increase in Basic Pay was Approved

If the head of the organizational unit approves the request, a data record containing the pay increase approved for the employee is created in the background for infotype 0008 *Basic Pay*.



If the plausibility and consistency check prevents the infotype record from being created, an error handling work item is sent to the inbox of the personnel administrator responsible for the employee.

By executing the work item, a new data record for infotype 0008 *Basic Pay* is presented to the personnel administrator for online processing. The approved data from the form is available as default values.

The error dialog of infotype 0008 *Basic Pay* helps the personnel administrator perform further processing.

Confirmation

If the infotype record is created, the employee's superior receives a mail in his or her inbox to confirm approval of the pay increase.

If the infotype record could not be created, the employee's superior and the head of the organizational unit are informed by mail. This concludes the workflow.

Informing the Employee

If the infotype record is created, the employee is informed by mail of the approved and implemented pay increase. The letter is not sent to the printer automatically. Instead, it is sent as a concluding work item to the appropriate personnel administrator's inbox. Executing the work item concludes the workflow.

Process Leave Request (PA-TIM-REC)

Use

The *Leave Request* workflow enables employees to enter a leave request in the Internet/intranet themselves. It could be a request for a vacation, for example. However, processes could also be implemented with which employees inform their superiors of the fact that they are unable to work.

The workflow is only ever used in conjunction with the *Leave Request* Internet application component. If you require further information, see [Leave Request \[Ext.\]](#).

Structure

The leave request is subject to an approval procedure, during which the employee's superior can check the data record to see if the employee's absence can be approved according to the schedule. If the record is approved but cannot be stored (because it collides with other records in the system, for example), the administrator responsible ensures that the leave request is returned to the employee.

Using & Linking to Functions (PA-TIM-REC)

The advantage of this workflow is that the persons responsible for approving or checking the data are informed automatically of the request, and can approve or reject it directly. If the request is rejected, the employee is informed. He or she can then respond accordingly. He or she can cancel the leave request, or change the dates.



The leave request services offered by the system include the [Cancel Leave Request \[Page 55\]](#) workflow. It enables employees to:

- Cancel or partially cancel a leave request.
- Delete unprocessed leave requests and cancelations.

The *Leave Request* workflow scenario is an excellent example of how employees can be involved in maintaining their own data, and how recorded data can be checked using an integrated approval procedure.

Using & Linking to Functions (PA-TIM-REC)

Enter Leave Request

The employee uses the enterprise's Internet/intranet to enter the leave request. If the requested leave is deducted from an absence quota (such as leave entitlement), the employee can use the service to check leave information directly. The employee can inform his or her superior of the reason for the absence by entering a note.

By choosing *Submit*, the request is forwarded to the superior.

Approve or Reject Leave Request

The user defined in the organizational plan as the superior of the person who enters the leave request receives a work item in his or her Internet/intranet inbox. The work item enables the superior to approve the leave request.

By executing the work item, the superior accesses the Internet service that enables him or her to view the current leave request data.

He or she can *approve* or *reject* the request. The reason for a rejection can be stated in a note.

Leave Request Rejected

If the leave request is not approved, the person who entered it receives a work item in his or her Internet/intranet inbox stating that the request was not approved. By executing the work item, the leave request is displayed again for the employee who submitted it. He or she can change or withdraw the leave request. If a change is made, the approval procedure is restarted.

Leave Request Approved

If the leave request is approved, an absence record (infotype 2001) is created in the background with the approved data of the requested absence.

Error Handling

If the plausibility and consistency check prevents the absence record from being created in the R/3 System, an error handling work item is sent to the administrator responsible. This work item

Preparation and Customizing (PA-TIM-REC)

is displayed in the Internet/intranet inbox and in the R/3 System. It can only be executed from the R/3 System inbox.

By executing the work item, the administrator opens the *Absences* infotype (2001) in *Create* mode. The approved data from the leave request is available as default values.

The administrator checks the situation to determine why the absence record could not be created. He or she is supported by the error dialog of infotype 2001. If the situation can be clarified, he or she approves the leave request. If it cannot be clarified because, for example, an absence already exists in the system for the period of this leave request, he or she rejects the leave request.

Confirmation

If a leave request is successfully posted, the employee who submitted it receives a message regarding the approved absence, along with any related information.

If posting is not possible, a message is sent to the employee who submitted the leave request and his or her superior.

Workflow is complete when this message is sent.

Preparation and Customizing (PA-TIM-REC)

You can use the following workflow template to define your own process:

Workflow Template for Leave Request

Name	Task
Leave Request With HTML	WS 20000081

In addition to general customizing, which ensures that the workflow system functions correctly, customizing is also required specifically for this workflow template.

Relate System User Name to Personnel Number

To ensure that an employee can be identified in workflow, a system user name must be related to the personnel number created for him or her.

The simplest way to define this relationship is by using infotype 0105 *Communication*.

Maintain infotype 0105 *Communication* for all of the employees allowed to enter their own absence data, and for all of the persons allowed to authorize this data. The system includes subtype/communication type **0001 - system user name (SY-UNAME)** for this purpose.



If you require further information, see the *Communication* sections of the Implementation Guide for Personnel Administration.

Set Up The Organizational Plan

The absence request is made by an employee, and approved or rejected by his or her superior. This is only possible if the relationship between the employee and the superior has been defined in the organizational plan.

The standard system includes the following role templates:

Role Templates

Role 168	Superior
Role 60100010	Administrator

Using an Organizational Plan to Assign Users

To define a suitable organizational plan for the scenario, access the Implementation Guide, choose *Basis Components* → *Business Management* → *SAP Business Workflow*, and perform the *Edit Organizational Plan* step.

Cancel Leave Request (PA_TIM_REC)

Use

The *Cancel Leave Request* workflow enables employees to perform the following activities in the Internet/intranet:

- Delete leave requests and cancelations that have not been processed by the person responsible for their approval
- Cancel an approved leave request if, for example, they do not take their leave
- Partially cancel an approved leave request if, for example, they want to shorten their leave by a few days

The workflow is only ever used in conjunction with the *Cancel Leave Request* Internet application component. If you require further information, see [Cancel Leave Request \[Ext.\]](#).

Structure

If an employee cancels an unprocessed leave request, the workflow is concluded and the work item concerned is deleted from the inbox of the person responsible for its approval.

The cancelation of the leave request is subject to an approval procedure, during which the employee's superior checks the data record to see if the cancelation can be approved according to the schedule.

Using & Linking to Functions (PA-TIM-REC)

The advantage of this workflow is that the persons responsible for approving the data are informed automatically of the cancelation, and can approve or reject it directly. In the event of approval or rejection, the employee is informed automatically and can respond accordingly. He or she can withdraw or change the cancelation.

Using & Linking to Functions (PA-TIM-REC)

Deleting Unprocessed Leave Requests and Cancelations

The employee uses the enterprise's Internet/intranet to delete leave requests and cancelations that have not been processed by the person responsible for their approval.

The work item is deleted, and the workflow is complete.

Enter Cancelation of Leave Request

The employee uses the enterprise's Internet/intranet to cancel a leave request that has already been approved. To facilitate data entry, the system displays a list of all approved absences from which the employee can choose the one that is required.

He or she can cancel the leave request completely, or perform a partial cancelation. A partial cancelation is only possible if the period in question is shortened.

By choosing *Cancel*, the request is forwarded to the superior.

Approve or Reject Canceled Leave Request

The user defined in the organizational plan as the superior of the person who enters the leave request receives a work item in his or her Internet/intranet inbox. The work item enables the superior to approve the leave request.

By executing the work item, the superior accesses the Internet service that enables him or her to view the current leave request data.

He or she can *approve* or *reject* the cancelation. The reason for a rejection can be stated in a note.

Cancelation of Leave Request Rejected

If the cancelation of the leave request is not approved, the person who entered it receives a work item in his or her Internet/intranet inbox stating that the cancelation was rejected. By executing the work item, the leave request is displayed again for the employee who submitted it. He or she can change or withdraw the cancelation. If a change is made, the approval procedure is restarted.

Cancelation of Leave Request Approved

If the leave request was approved, the absence record in question (infotype 2001) is deleted in the background.

Preparation and Customizing (PA-TIM-REC)

Error Handling

If the cancelation cannot be posted in the R/3 System, the administrator responsible receives an error handling work item. This work item is displayed in the Internet/intranet inbox and in the R/3 System. It can only be executed from the R/3 System inbox.

By executing the work item, the administrator opens the *Absences* infotype (2001) in *Change* mode. The approved data from the leave request is available as default values.

The administrator checks the situation to determine why the absence record could not be deleted or changed. He or she is supported by the error dialog of infotype 2001. If the situation can be clarified, he or she approves the cancelation. If it cannot be clarified, he or she rejects the cancelation.

Confirmation

If an absence is successfully changed or deleted, the employee who submitted it receives a message regarding the approved cancelation, along with any related information.

If posting is not possible, a message is sent to the employee who submitted the absence request and his or her superior.

Workflow is complete when this message is sent.

Preparation and Customizing (PA-TIM-REC)

You can use the following workflow template to define your own process:

Workflow Template for Cancelation of Leave Request

Name	Task
Cancelation of Leave Request	WS 0100109

In addition to general customizing, which ensures that the workflow system functions correctly, customizing is also required specifically for this workflow template.

Relate System User Name to Personnel Number

To ensure that an employee can be identified in workflow, a system user name must be related to the personnel number created for him or her.

The simplest way to define this relationship is by using infotype 0105 *Communication*.

Maintain infotype 0105 *Communication* for all of the employees allowed to enter their own absence data, and for all of the persons allowed to authorize this data. The system includes subtype/communication type **0001 - system user name (SY-UNAME)** for this purpose.



If you require further information, see the *Communication* sections of the Implementation Guide for Personnel Administration.

Set Up The Organizational Plan

Set Up The Organizational Plan

The absence request is made by an employee, and approved or rejected by his or her superior. This is only possible if the relationship between the employee and the superior has been defined in the organizational plan.

The standard system includes the following role templates:

Role Templates

Role 168	Superior
Role 60100010	Administrator

Using an Organizational Plan to Assign Users

To define a suitable organizational plan for the scenario, access the Implementation Guide, choose *Basis Components* → *Business Management* → *SAP Business Workflow*, and perform the *Edit Organizational Plan* step.

Workflow Scenarios in Travel Management (FI-TV)

Definition

The workflow included in the *Travel Management* application component supports the approval procedures for

- Travel planning
- Travel request
- Reimbursement of expenses after a business trip

Use

The standard system includes the following workflow templates for Travel Management:

- [Approve Travel Request \[Page 59\]](#)
- [Approve Travel Plan \[Page 60\]](#)
- [Approve Trip \[Page 62\]](#)

Approve Travel Request

Process Flow

The *approve travel request* workflow describes the following process:

The trip data is recorded in the system by the employee.

The employee's superior is responsible for approving the trip. The system uses the organizational plan to determine the superior automatically.

The superior can approve the trip, send it back to the employee for correction, or reject it.

If the superior approves the trip, the employee who submitted the request is informed of this approval in an automatically generated mail.

If corrections are necessary, the superior creates a message as a workflow attachment. The employee receives a work item with this attachment. When this work item is performed, the transaction for changing the trip is started.

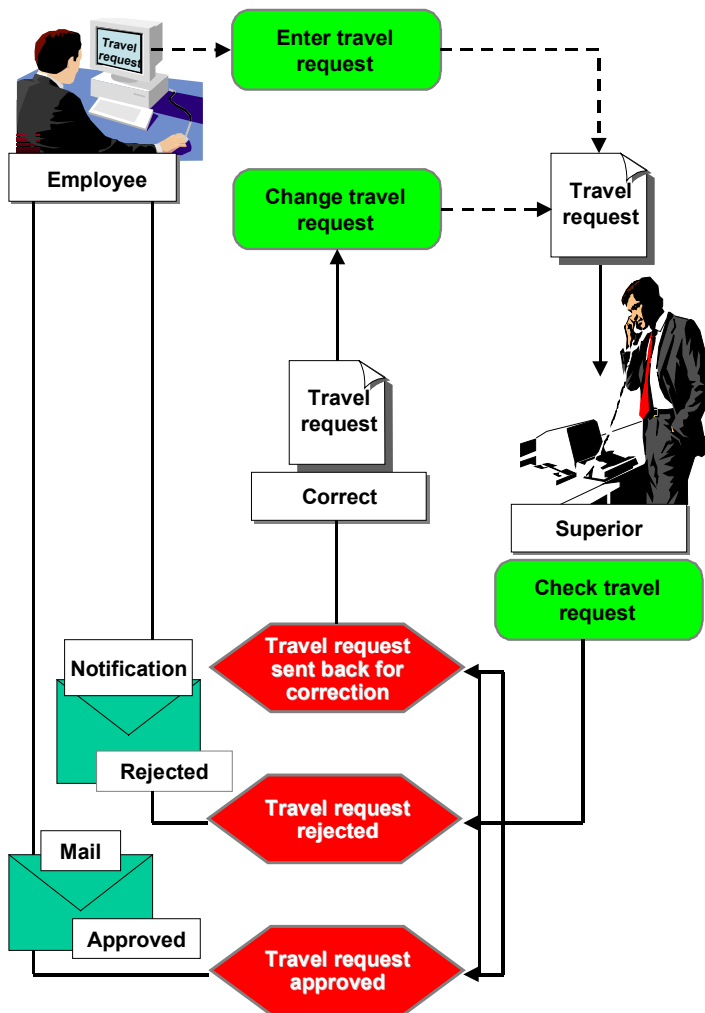
If the travel request is rejected, the superior creates a message that is sent to the employee as a mail.

See also:

[Graphic: Approve Travel Request \[Page 59\]](#)

Graphic: Approve Travel Request

Approve Travel Plan



Approve Travel Plan

Process Flow

The [Approve Travel Plan \[Ext.\]](#) workflow describes the following process:

1. The travel planning data is recorded in the system by the employee.
2. The superior of the employee embarking on the business trip is responsible for approving the travel plan. The system uses the organizational plan to determine the superior automatically.

The superior can approve the travel plan, send it back to the employee embarking on the business trip for correction, or reject it.

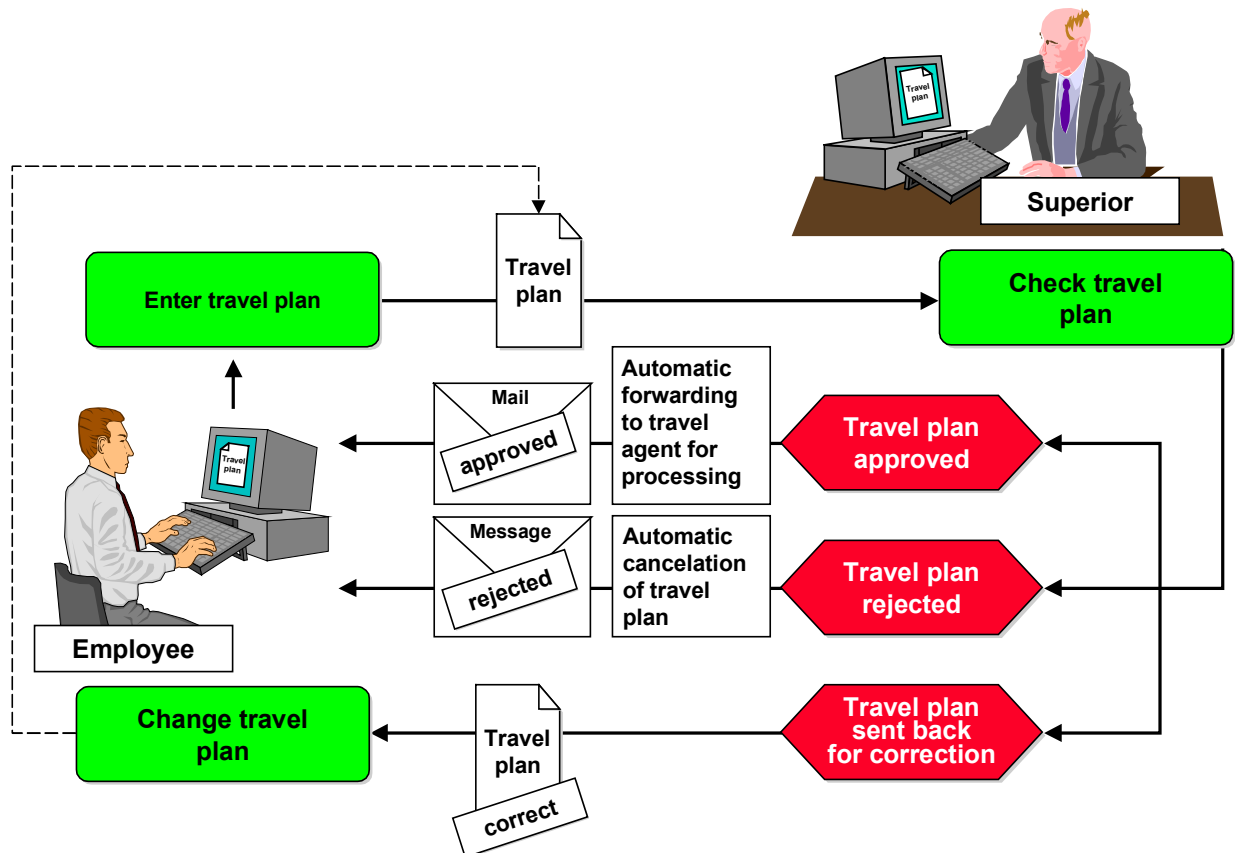
Approve Travel Plan

- **Approval:** If the superior approves the travel plan, the employee embarking on the business trip is informed of this approval in an automatically generated mail. The [posting record \(PNR\) \[Ext.\]](#) with the reservation information included in the travel plan is written to a processing queue for the connected travel agent.
- **Correction:** If corrections are necessary, the superior creates a message as a workflow attachment. The employee receives a work item with this attachment. When this work item is performed, the transaction for changing the travel plan is started.
- **Rejection:** If the travel plan is rejected, the superior creates a message that is sent to the employee embarking on the business trip as a mail. The posting procedure for travel services included in the travel plan is canceled.

Graphic: Approve Travel Plan

Graphic: Approve Travel Plan

Approve Trip



Approve Trip

Process flow

The *approve trip* workflow describes the following process:

After completing a trip, the employee supplements the trip data in the system.

The expenses department checks the trip data to see if it is plausible and if it complies with the statutory or enterprise-specific trip provisions. Its consistency with original (or facsimile) receipts is also verified. The trip can be forwarded to the superior, or sent back to the employee for correction.

If corrections are necessary, an employee in the expenses department creates a message as a workflow attachment. The employee who entered the trip data receives a work item with this attachment. To change the trip, the employee can execute the appropriate transaction directly from the integrated inbox. When the trip is saved, the workflow is started again.

Graphic: Approve Trip

If the trip data is consistent, it is then checked by the superior. The superior can approve the trip, send it back to the employee for correction, or reject it.

If corrections are necessary, the superior creates a message as a workflow attachment. The employee receives a work item with this attachment, and can access the transaction to change the trip directly.

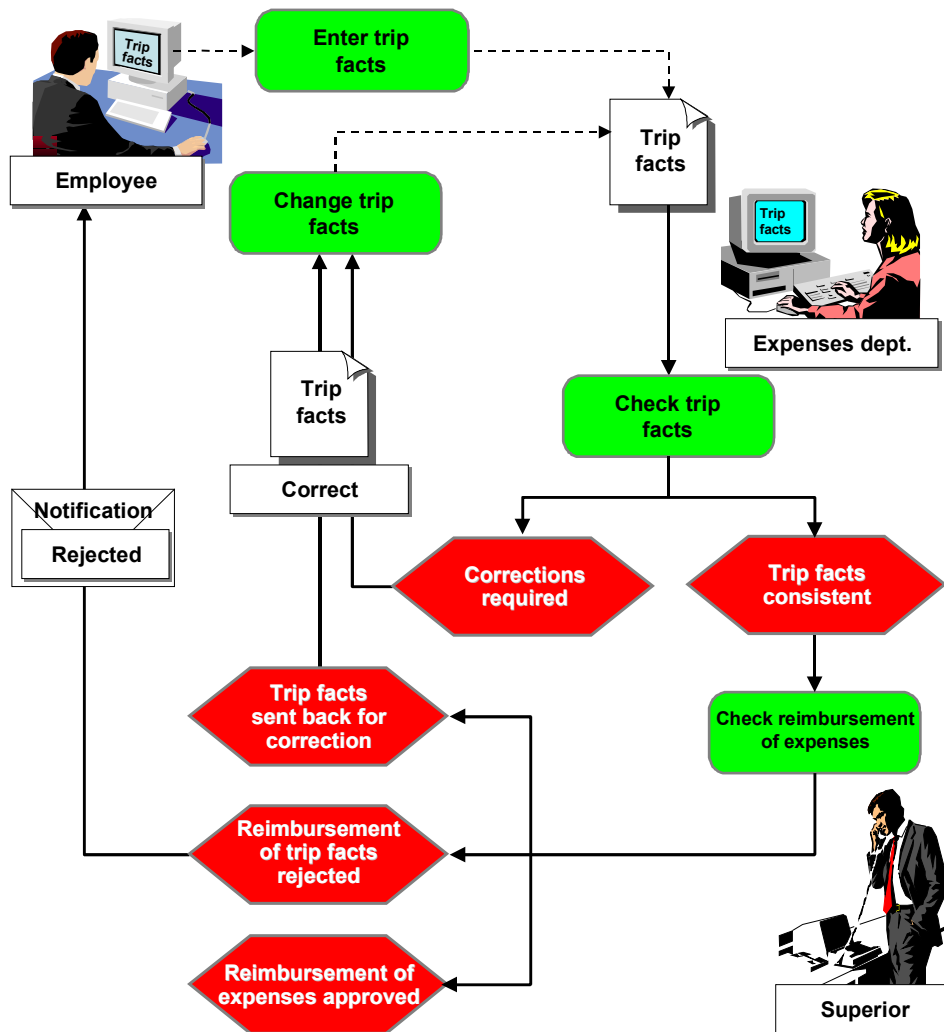
If the trip is rejected, the superior creates a message that is sent to the employee as a mail.

See also:

[Graphic: Approve Trip \[Page 62\]](#)

Graphic: Approve Trip

Technical Background (FI-TV)



Technical Background (FI-TV)

Object Types Used

The interface between R/3 functions and workflow is made possible by object technology.

The following information is of a technical nature. You require this information if you are interested in implementation details or want to perform your own enhancements.

[Object Type BUS2089 \(Employee Trip\) \[Page 65\]](#)

[Object Type SELFITEM \(Work Item for Dialog Steps on Self\) \[Page 65\]](#)

Standard Tasks

Standard tasks, which are supplied by SAP as single-step tasks, describe elementary business activities from an organizational perspective. In each case, a single-step task relates to one object method (that is, a technical link to R/3 functions). It is related to persons in the organization who could be its agents.

[Standard Task TS20000118 \(Approve Travel Request\) \[Page 66\]](#)

[Standard Task TS20000130 \(Set Trip Status to Approved\) \[Page 66\]](#)

[Standard Task TS20000095 \(Send Mail: Request Approved\) \[Page 67\]](#)

[Standard Task TS01000124 \(Approve Travel Plan\) \[Page 67\]](#)

[Standard Task TS01000126 \(Send Mail: Travel Plan Approved\) \[Page 68\]](#)

[Standard Task TS60300084 \(Change Status of Travel Plan\) \[Page 68\]](#)

[Standard Task TS60300085 \(Write PNR to Processing Queue\) \[Page 69\]](#)

[Standard Task TS01000129 \(Change Travel Plan\) \[Page 69\]](#)

[Standard Task TS60300086 \(Cancel Travel Plan\) \[Page 70\]](#)

[Standard Task TS20000189 \(Change Trip\) \[Page 70\]](#)

[Standard Task TS20000139 \(Enter and Send Short Message\) \[Page 71\]](#)

[Standard Task TS20000135 \(Check Trip Facts\) \[Page 71\]](#)

[Standard Task TS20000131 \(Approve Trip Facts\) \[Page 72\]](#)



User Decisions

Standard tasks *TS01000124 (Approve Travel Plan)*, *TS20000118 (Approve Travel Request)*, *TS20000135 (Check Trip)*, and *TS20000131 (Approve Trip Facts)* are user decisions. Standard task *TS00008267 (Generic Decision Task)* was used as a template.

When these tasks are performed, the descriptive long text for the task is displayed. The long text contains text variables that are filled at runtime with the attribute values of object type *BUS2089 (Employee Trip)*.

See also:

[Tasks and Task Groups \[Ext.\]](#)

Roles

A role is used to determine the superior assigned to the employee embarking on the business trip.

[Manager Role \(Superior Of...\) \[Ext.\]](#)

See also:

[Roles \[Ext.\]](#)

Object Type BUS2089 (Employee Trip)

Workflow Template

The actual process is implemented as a workflow template. Your R/3 System includes the following workflow templates:

[Workflow Template WS20000050 \(Approve Travel Request\) \[Page 72\]](#)

[Workflow Template WS 01000087 \(Approve Travel Plan\) \[Page 73\]](#)

[Workflow Template WS20000040 \(Approve Trip\) \[Page 74\]](#)

Object Type BUS2089 (Employee Trip)

In this scenario, the system uses a business application object of type **BUS2089 (Employee Trip)**, which means that an employee's trip data is processed.

Events

When a trip is saved in the database, events of object type **BUS2089** are created depending on the status of the travel plan or travel expenses.

Event creation for the *Approve Travel Plan* workflow is implemented in the `FITP_TRIGGER_WORKFLOW` function module.

Event creation for the *Approve Travel Request* and *Approve Trip* workflows is implemented as `TRIGGER_WORKFLOW` in the `FITP_WORKFLOW` include.



In the case of batch input, event creation cannot be guaranteed.

Position of the Object Type in the Business Object Repository:

Financial Accounting → Travel Management

OT SELFITEM (Work Item for Dialog Steps on Self)

Mail dispatch has been technically implemented so that the long text of a single-step task is sent. This long text is entered in the definition of a single-step task.

This means that a send method must be available that operates on the work item itself, which represents the single-step task at runtime. The **SELFITEM** object type (*work item for dialog steps on self*) provides methods that operate "on itself".

Position of the Object Type in the Business Object Repository:

Basis → Business Management → Workflow

Standard Task TS20000118 (Approve Travel Request)

Standard Task TS20000118 (Approve Travel Request)

When this standard task is performed, a travel request can be approved, sent back to the employee for correction, or rejected.

Standard task: TS20000118

Abbreviation: ApproveReq

Name: *Approve Travel Request*

Referenced Object Method and Characteristics

Object type: DECISION (*selection*)

Method: Process (*execute*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee's superior via *role resolution*.

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the single-step task. For this reason, you must relate the standard *approve travel request* task to a job called "head of department" (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS20000130 (Set Trip Status to Approved)

In this standard task, the status of a travel request/completed trip is set to *approved*.

Standard task: TS20000130

Abbreviation: TripStatAppr

Name: *Set Trip Status to Approved*

Referenced Object Method and Characteristics

Object type: BUS2089 (*EmployeeTrip*)

Method: Approve

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee's superior via *role resolution*.

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the single-step task. For this reason, you must relate the

Standard Task TS20000095 (Send Mail: Request Approved)

standard set *trip status to approved* task to a job called “head of department” (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS20000095 (Send Mail: Request Approved)

When this standard task is performed, the descriptive long text is sent to the employee.

Standard task: TS20000095

Abbreviation: MailReqAppr

Name: *Send Mail: Request Approved*

Referenced Object Method and Characteristics

Object type: SELFITEM (*work item for dialog steps on self*)

Method: SendTaskDescription

Characteristics: none

Maintain Agent Assignment

You do not need to assign agents because the standard task runs in the background.

Standard Task TS01000124 (Approve Travel Plan)

When this standard task is performed, a travel plan can be approved, sent back to the employee who entered it for correction, or rejected.

Standard task: TS01000124

Abbreviation: ApprTripPlan

Name: *Approve Travel Plan*

Referenced Object Method and Characteristics

Object type: DECISION (*technical implementation of a user decision*)

Method: Process (*perform*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee’s superior via [role resolution \[Ext.\]](#).

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the [single-step task \[Ext.\]](#). For this reason, you must relate

Standard Task TS01000126 (Send Mail: Travel Plan Approved)

the standard *Approve Travel Plan* task to a job called “head of department” (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS01000126 (Send Mail: Travel Plan Approved)

When this standard task is performed, the description of the task is sent to the employee.

Standard task: TS01000126

Abbreviation: MailPlanAppr

Name: *Send Mail: Travel Plan Approved*

Referenced Object Method and Characteristics

Object type: SELFITEM (*work item for dialog steps on self*)

Method: SendTaskDescription (*send description of task*)

Characteristics: none

Maintain Agent Assignment

You do not need to assign agents because the standard task runs in the background.

Standard Task TS60300084 (Change Status of Travel Plan)

This standard task changes the approval status of a travel plan.

Standard task: TS60300084

Abbreviation: TripStatAppr

Name: *Change Status of Travel Plan*

Referenced Object Method and Characteristics

Object type: BUS2089 (*Employee Trip*)

Method: ChangePlanStatus (*Change Approval Status of Travel Plan*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee’s superior via [role resolution \[Ext.\]](#).

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the [single-step task \[Ext.\]](#). For this reason, you must relate the standard *Change Approval Status of Travel Plan* task to a job called “head of department” (or

Standard Task TS60300085 (Write PNR to Processing Queue)

similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS60300085 (Write PNR to Processing Queue)

This standard task writes the [posting record \(PNR\) \[Ext.\]](#) that is assigned to the [travel plan \[Ext.\]](#) to a processing queue for the travel agent.

Standard task: TS60300085

Abbreviation: SetPlanStatu

Name: *Write PNR to Processing Queue*

Referenced Object Method and Characteristics

Object type: BUS2089 (*Employee Trip*)

Method: QueuePNR (*Write PNR to Processing Queue*)

Characteristics: none

Maintain Agent Assignment

You do not need to assign agents because the standard task runs in the background.

Standard Task TS01000129 (Change Travel Plan)

When this standard task is performed, the transaction for changing the [travel plan \[Ext.\]](#) is called up.

Standard task: TS01000129

Abbreviation: ChangePlan

Name: *Change Travel Plan*

Referenced Object Method and Characteristics

Object type: BUS2089 (*Employee Trip*)

Method: ChangePlan (*Change Travel Plan (TP01)*)

Characteristics: *Confirm End of Processing*

Maintain Agent Assignment

At runtime, this standard task is executed by the person who entered the trip data. This could be any employee, so you must classify the [single-step task \[Ext.\]](#) as a *general task* in Customizing.

Standard Task TS60300086 (Cancel Travel Plan)

Standard Task TS60300086 (Cancel Travel Plan)

This standard task cancels all [posting record \(PNR\) \[Ext.\]](#) segments.

Standard task: TS60300086

Abbreviation: CancelPlan

Name: *Cancel Travel Plan*

Referenced Object Method and Characteristics

Object type: BUS2089 (*Employee Trip*)

Method: CancelPlan (*Cancel Travel Plan*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee's superior via [role resolution \[Ext.\]](#).

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the [single-step task \[Ext.\]](#). For this reason, you must relate the standard *Cancel Travel Plan* task to a job called "head of department" (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS20000189 (Change Trip)

When this standard task is performed, the transaction for changing the trip is called up.

Standard task: TS20000189

Abbreviation: ChangeTrip

Name: *Change Trip*

Referenced Object Method and Characteristics

Object type: BUS2089 (*Employee Trip*)

Method: ChangeInteractive (*change trip (interactive)*)

Characteristics: *confirm end of processing*

Maintain Agent Assignment

At runtime, this standard task is executed by the person who entered the trip data. This could be any employee, which means that this single-step task must be classified as a *general task* in Customizing.

Standard Task TS20000139 (Enter and Send Short Message)

Standard Task TS20000139 (Enter and Send Short Message)

When this standard task is performed, the transaction for entering and sending short messages is called up.

Standard task: TS20000139

Abbreviation: SendText

Name: *Enter and Send Short Message*

Referenced Object Method and Characteristics

Object type: SOFM (*office document*)

Method: Compose (*create and send*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee's superior via *role resolution*.

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the single-step task. For this reason, you must relate the standard *enter and send short message* task to a job called "head of department" (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Standard Task TS20000135 (Check Trip Facts)

When this standard task is performed, the basic trip data is displayed. The trip can be released for approval, or sent back to the employee for correction.

Standard task: TS20000135

Abbreviation: CheckTrip

Name: *Check Trip Facts*

Referenced Object Method and Characteristics

Object type: DECISION (*selection*)

Method: Process (*execute*)

Characteristics: none

Maintain Agent Assignment

When customizing this task, you must maintain agent responsibilities so that all employees in the expenses department are possible agents of this task.

Standard Task TS20000131 (Approve Trip Facts)

When this standard task is performed, the basic trip data is displayed. The trip facts can be approved, sent back for correction, or rejected.

Standard task: TS20000131

Abbreviation: ApproveTrip

Name: *Approve Trip Facts*

Referenced Object Method and Characteristics

Object type: DECISION (*selection*)

Method: Process (*execute*)

Characteristics: none

Maintain Agent Assignment

At runtime, this standard task is addressed to the employee's superior via *role resolution*.

When customizing this task, you must keep the agent assignment so general that all superiors of any employee are possible agents of the single-step task. For this reason, you must relate the standard *approve trip facts* task to a job called "head of department" (or similar), and maintain the organizational plan so that the respective chief positions of each organizational unit are described by this job.

Template WS20000050 (Approve Travel Request)

If a trip is created in the system with the *travel request entered/to be accounted* status, a workflow is started based on the RequestAppr template.

Workflow template: WS20000050

Abbreviation: RequestAppr

Name: *Approve Travel Request*

Triggering Event of Workflow Template

The Requested event (*travel request to be approved*) of the BUS2089 object type (*employee trip*) is entered as the triggering event of the workflow template.

Workflow Template WS 01000087 (Approve Travel Plan)

This “linkage” between the event and the workflow template to be started is deactivated in the standard system. If you want to start the workflow template, the linkage must be activated in Customizing for SAP Business Workflow.

Workflow Container and Binding

The most important information that must be available during workflow are the object reference to the trip to be processed (`_Evt_Object`), and the name of the person who entered the trip data (`_Evt_Creator`). This information is available as event parameters in the container for the triggering event, and must be transferred to the workflow container by “binding”.

The standard system includes the following binding definition between the triggering event and the workflow container:

<u>Workflow Container</u>		<u>Event Parameter Container</u>
<code>_WF_Initiator</code>	<-	<code>_Evt_Creator</code>
<code>EmployeeTrip</code>	<-	<code>_Evt_Object</code>

In the standard system, the `_WF_Initiator` element is included in the workflow container. The `EmployeeTrip` element was also created in addition to the existing standard elements.

Workflow Template WS 01000087 (Approve Travel Plan)

If a new or changed travel plan is saved/posted, a workflow of the `PlanApproval` template is started.

Workflow template: WS01000087

Abbreviation: `PlanApproval`

Name: *Approve Travel Plan*

Triggering Event of Workflow Template

The `PlanCreated` event (*Travel Plan Created*) of the `BUS2089` object type (*Employee Trip*) is entered as the triggering event of the workflow template.



This “linkage” between the event and the workflow template to be started is deactivated in the standard system. If you want to start the workflow template, the linkage must be activated in Customizing for SAP Business Workflow.

Workflow Container and Binding

The most important information that must be available during workflow are the object reference to the trip to be processed (`_Evt_Object`), and the name of the person who entered the trip data

Workflow Template WS20000040 (Approve Trip)

(**_Evt_Creator**). This information is available as event parameters in the container for the triggering event, and must be transferred to the workflow container by “binding”.

The standard system includes the following binding definition between the triggering event and the workflow container:

Workflow Container		Event Parameter Container
_WF_Initiator	←	_Evt_Creator
EmployeeTrip	←	_Evt_Object

In the standard system, the **_WF_Initiator** element is included in the workflow container. The **EmployeeTrip** element was also created in addition to the existing standard elements.

Workflow Template WS20000040 (Approve Trip)

When a trip is created with the *trip completed/to be accounted* status, the *approve trip* workflow is started.

Workflow template: WS20000040

Abbreviation: TripApproval

Name: Approve Trip

Triggering Event of Workflow Template

The **Created** event (*trip to be approved*) of the **BUS2089** object type (*employee trip*) is entered as the triggering event of the workflow template.



This “linkage” between the event and the workflow template to be started is deactivated in the standard system. If you want to start the workflow template, the linkage must be activated in Customizing for SAP Business Workflow.

Workflow Container and Binding

The most important information that must be available during workflow are the object reference to the trip to be processed (**_Evt_Object**), and the name of the person who entered the trip data (**_Evt_Creator**). This information is available as event parameters in the container for the triggering event, and must be transferred to the workflow container by “binding”.

The standard system includes the following binding definition between the triggering event and the workflow container:

<u>Workflow Container</u>		<u>Event Parameter Container</u>
_WF_Initiator	<-	_Evt_Creator
EmployeeTrip	<-	_Evt_Object

In the standard system, the **_WF_Initiator** element is included in the workflow container. The **EmployeeTrip** element was also created in addition to the existing standard elements.

Preparation and Customizing (FI-TV)

Use

You can use the following workflow templates when defining your process.

<u>Name</u>	<u>Task</u>
Approve Travel Request	WS 20000050
Approve Travel Plan	WS 01000087
Approve Trip	WS 20000040

In addition to general customizing for *Travel Management* and workflow, specific customizing steps are required for this workflow template.

[Relationship Between System User Name and Personnel Number \[Page 75\]](#)

[Set Up the Organizational Plan \[Page 76\]](#)

[Task-Specific Customizing \[Page 76\]](#)

[Maintenance of Event Receiver Linkage \[Page 77\]](#)

Relationship Between System User Name & Pers. No.

Use

To ensure that an employee can be identified in workflow, a system user name must be related to the personnel number created for him or her.

The simplest way to define this relationship is by using infotype 0105 *Communication*.

Activities

Maintain the *Communication* infotype (0105) for all of the employees who are possible agents for *Travel Management* workflows. The system includes subtype/communication type **0001 - system user name (SY-UNAME)** for this purpose.



If you require further information, see the relevant sections in the Implementation Guide by choosing *Personnel Management* → *Personnel Administration* → *Communication*.

Defining the Organizational Plan

Use

All of the persons who use the *Travel Management* workflow must be identified in Customizing for SAP Business Workflow.

Activities

To define a suitable organizational plan for the scenario, access the Implementation Guide, choose *Basis Components* → *Business Management* → *SAP Business Workflow*, and perform the *Edit organizational plan* step.

Important: Define positions created for the superior as *chief positions*.

Task-Specific Customizing

Use

Possible agents must be specified for every standard task used in the *Travel Management* workflows. By doing this, you define the organizational assignment for processing.

Activities

Call up the Implementation Guide, choose *Basis Components* → *Business Management* → *SAP Business Workflow*, and perform the *Perform Task-Specific Customizing* step.

- If you use the *Approve Travel Plan* workflow:
 - a. Choose *Financial Accounting* → *Travel Management* → *Travel Planning* → *Assign Tasks to Agent*.
 - b. Assign agents for the following standard tasks:

TS 01000124	Approve Travel Plan
TS 60300084	Change Status of Travel Plan
TS 01000129	Change Travel Plan
TS 60300086	Cancel Travel Plan

- If you use the *Approve Travel Request* and/or *Approve Trip* workflows:
 - a. Choose *Financial Accounting* → *Travel Management* → *Trip Costs* → *Assign Tasks to Agent*.
 - b. Assign agents for the following standard tasks:

TS 20000118	Approve Travel Request
TS 20000130	Set Trip Status to Approved

Maintenance of Event Receiver Linkage

TS 20000131	Approve Trip Facts
TS 20000135	Check Trip Facts
TS 20000139	Create and Send Text
TS 20000189	Change Trip

Maintenance of Event Receiver Linkage

Use

To ensure that the workflow templates for *Travel Management* are started, linkage between the triggering event and the workflow template as the receiver of the event must be activated in Customizing for SAP Business Workflow.

Activities

Call up the Implementation Guide, choose *Basis Components* → *Business Management* → *SAP Business Workflow*, and perform the *Perform Task-Specific Customizing* step.

- If you use the *Approve Travel Plan* workflow:

Choose *Financial Accounting* → *Travel Management* → *Trip Planning* to access the application component, and activate event linkage for the *Approve Travel Plan* workflow template.

- If you use the *Approve Travel Request* and/or *Approve Trip* workflows:

Choose *Financial Accounting* → *Travel Management* → *Trip Costs* to access the application component, and activate event linkage for the *Approve Travel Request* and/or *Approve Trip* workflow templates.



Alternatively, you can activate event receiver linkage by processing workflow templates **WS01000087** (*Approve Travel Plan*), **WS20000050** (*Approve Travel Request*), and **WS20000040** (*Approve Trip*) directly.

Using and Linking to Application Functions (FI-TV)

Purpose

The *Approve Travel Plan*, *Approve Travel Request*, and *Approve Trip* workflows support the approval procedure in *Travel Management*.

Process Flow

Workflow is started by an event that is created when a trip is saved.

Automatic Generation of Documents for HR Applications

The *Approve Travel Plan* workflow is started each time a travel plan is saved/posted.

The *Approve Travel Request* workflow is started when a trip is saved with the *travel request/to be accounted* status.

The *Approve Trip* workflow is started when a trip is saved with the *trip completed/to be accounted* status.

Changing Trip Data Without Workflow

If trip data is changed (using a data entry transaction, for example) after workflow has been started, the workflow is canceled automatically when the trip is saved. If the trip has the appropriate status, a new workflow is started at the same time.

This means that data entry transactions can still be used to approve a travel plan, travel request, or completed trip when a workflow is active. You can also approve travel requests and completed trips using the [approval program \[Ext.\]](#) in *Travel Management*.

Automatic Generation of Documents for HR Applications

Definition

This documentation describes the automatic generation of documents for HRMS applications using SAP Business Workflow.

It is a guide to creating documents using R/3 SAPscript functions, and defines all of the steps needed to set up a document for creating a workflow.

Please note that this document is only a brief overview. If you require detailed information, see [Workflow Scenarios in the Applications \[Ext.\]](#).

Use

Within *Personnel Administration*, you need to print a considerable number of letters and documents. The content of these documents is closely related to master data stored in the R/3 System:

- Letters sent to the employee, such as invitations and notifications
- Notifications sent to third parties, such as social insurance funds, tax offices, or health insurance companies
- Internal documents about changes to employee master data, such as illness, strikes, and maternity protection
- Legal documents, such as contracts, forms, and requests

A system used to create documents must be able to print any documents triggered by a change to employee master data.

Prerequisites

The system must be able to

Design Implementation in SAP Business Workflow®

- Print documents as soon as a master data field is changed. This function is necessary if you only want to print a few documents for a small group of employees.
- Store information on changes to master data, and then print the documents later. This function is particularly important if you want to print a large number of documents.

In addition, the system settings must enable you to assign more than one document to a specific master data change. If an employee falls sick, for example, a notification must be printed for the HR department and for the health insurance company.

If you want to generate documents, use the output media that are supported by SAPscript. SAPscript currently supports the following media: screen, printer, telefax, and telex.

See also:

[Design Implementation in SAP Business Workflow® \[Page 79\]](#)

Design Implementation in SAP Business Workflow®

A document created by workflow as a result of changes to master data can be illustrated as follows:

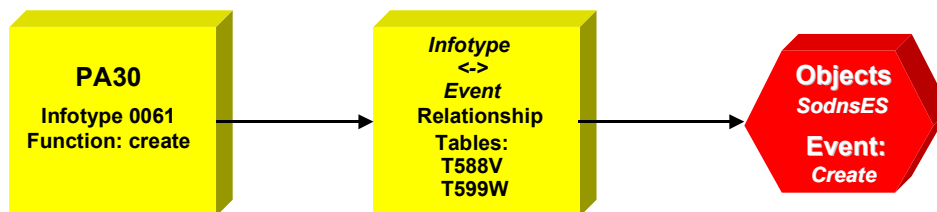


Figure: Document Generation in HRMS Applications using Workflow

The implementation of automatic document creation consists of the following steps:

- Define the form and text using SAPscript
- Define the required business objects in the Business Object Repository (BOR)
- Set up event generation
- Define standard tasks
- Define workflow

Technical Background

[Business Objects \[Page 80\]](#)

[Event Generation \[Page 81\]](#)

[HR Status Management \[Page 81\]](#)

[Define Standard Tasks \[Page 82\]](#)

[Object Method \[Page 83\]](#)

[Container Elements \[Page 83\]](#)

[Role \[Page 84\]](#)

[Example: Standard Task for Printing a Contract \[Page 84\]](#)

Business Objects

You must define employee data required in the generated document as business object attributes. Several business objects have already been implemented in the HR module, such as personal data, addresses, basic pay, and recurring payments/deductions. If you require country-specific documentation, you can use the tools included in the workbench. This procedure is well documented.

Within the text, data can be accessed as text symbols.

&ObjectType.Attribute.Attribute&



You require the employee's address.

Object type: **PersonnelData**

Attribute: **StreetAndHouseNumber**

Within the text, the data is accessed as follows:

&PersonnelData.StreetAndHouseNumber&



Use business object **STD_TEXT** to access the data as text symbols.

If the referenced attribute is an object, you can use a third level reference. For example, the personnel country grouping can be referred to as follows:

&Employee.PersonnelArea.CountryGrouping&

Event Generation



Object **STD_TEXT** only accepts references up to the third level.

One of the main advantages of using this object-oriented technique is that it enables you to access employee data intuitively.

If you require further information, see [Business Object Builder \[Ext.\]](#).

Event Generation

Use

Workflow is always triggered by an event. For this reason, the application must create the correct events so that it can start a document generation workflow.

There are several ways to create events:

1. The application calls up the appropriate function module (that is, **SWE_EVENT_CREATE**).
2. Events are created automatically for objects that use general status management.
3. Changes of status for objects logged as change documents.

The HRMS module uses special status management for objects that are related to infotypes. At this time, event generation triggered by change documents is not used in HRMS.

Events can be created on the basis of HR reports and programs by calling up **SWE_EVENT_CREATE** directly.

See also:

[HR Status Management \[Page 81\]](#)

HR Status Management

Use

Event generation within the HR module uses two different relationships:

1. Infotype table <-> object type (T588V)
2. Infotype table, operation <-> event, (function module) (T588W)

The former is a simple relationship between an infotype and a business object. The latter facilitates flexible event generation, depending on the operation performed by a particular infotype.

Define Standard Tasks

If you need more complex rules to create a specific event after a particular operation, you can specify the name of a function module responsible for event generation. The name

HR_EVENT_RULES_PA0001 is based on the following convention:

HR_EVENT_RULES_PYXXXX

XXXX	=	Infotype number
Y	=	A Personnel Administration B Recruitment

The module interface has a fixed structure (see HR_EVENT_RULES_PA0000).

Define Standard Tasks

The aim of automatic document generation is to create a printed version of a specific text. For this reason, the print function must be defined in the workflow used.

Each standard task in the system is based on the object method. Therefore, an appropriate object is required for printing texts.

Standard task definition is comprised of the following steps:

- Find the appropriate object method for printing a text.
- Define the standard task (*Tools → SAP Business Workflow → Development → Definition tools → Tasks*).
- Reference the appropriate object method.
- Define container elements.
 - Define a triggering event.



This step is usually required if the standard task is triggered by an event. The standard task is normally used within a workflow that has its own triggering events. If this is the case, you are not required to define a triggering event for the task.

- Define the binding between the event and task container.
- Define the employee responsible for the task.

See also:

[Object Method \[Page 83\]](#)

[Container Elements \[Page 83\]](#)

[Role \[Page 84\]](#)

[Example: Standard Task for Printing a Contract \[Page 84\]](#)

Object Method

Object Method

The BOR includes object type **STD_TEXT** for editing, displaying, and printing SAPscript texts. This object type is maintained by BASIS.

The following table illustrates object type **STD_TEXT**.

Object type		STD_TEXT
Key field:	Name	Text name (70 characters)
Methods:	ExistenceCheck	Check existence of SAPScript document
	Create	Create a new document
	ReplaceAndPrintBatch	Print document in batch with replacement of symbols
	ReplaceAndPrint	Replace text symbols and print document (with text display)
	Delete	Delete document
	Edit	Change document
	Display	Display document with text symbols
	ReplaceAndDisplay	Replace text symbols and display document

Figure: Business Object STD_TEXT: Methods

Container Elements

The main advantage of the **STD_TEXT** object is that it can print container elements as text symbols.

Each container element can be printed as follows:

```
&ContainerElement&
```

If the container element is an object, each attribute can be printed as follows:

&Object.Attribute&

If you want to print employee data, a container element of type *EmployeeT* must be included. This element enables each employee data record (defined as an attribute of *EmployeeT*) to be printed. If the documents are country-specific, it is a good idea to define the container element as *EmployeeXX* (ISO country key = personnel country grouping), which includes all of the international and country-specific attributes.

STD_TEXT can make distinctions between object attributes up to a third level, which means you must define utility objects in the container if higher levels are needed to make more distinctions.



STD_TEXT cannot make distinctions for the following expression:

Employee.PersonnelArea.LegalPerson.ActivityText.

For this reason, you must define a utility container element of the type *PersonnelArea*:

PersonnelArea<==Employee.PersonnelArea

Using this container element, the *activitytext* attribute can be printed in the text as follows:

PersonnelArea.LegalPerson.ActivityText.

If the document requires additional information from other object types, you must define the elements as container elements in the standard task.

Role

Each standard task can be assigned a standard role, which acts as a default role.

Within workflow, the various work items must be processed by specific employees. The name (or names) of the employee responsible is determined by the role definition.

The HRMS module includes the standard *personnel officer* role (standard role 400135, menu path *Tools → SAP Business Workflow → Development → Definition tools → Standard roles*).

This role is based on the **PERSCLERK** object type, which is the business object representation of table T526.

In practice, customers define their own roles and relate them to the HRMS organization model. The use of function modules is recommended for complex role definitions.

Example: Standard Task for Printing a Contract

Object type: **STD_TEXT**

Method: **ReplaceAndPrint**

Workflow Definition

Container elements:

- **EmployeeES:**
Contains all international infotypes, and country-specific infotype 0061 (Social Insurance).
- **PersonnelArea:**
Enables you to make distinctions between up to more than three levels.
- **WageTypePosition_1:**
First wage type in infotype 0008.
- **WageTypePosition_2:**
Second wage type in infotype 0008.



The following international infotypes exist: 0001; 0002; 0005; 0006; 0007; 0008; 0014; and 2001.



The standard task to which the example refers is used in workflow. It is not started independently. If you require more information on event processing and binding, see [Workflow Definition \[Page 85\]](#).

Workflow Definition

To specify **when** the print procedure starts, you define a triggering event. To specify **how** a document is printed, you define a standard task. To specify **which** document is printed, and the **circumstances under which** the document is printed, you use the workflow definition to formulate complex rules.

A document is usually printed in accordance with particular country-specific rules, which means that the description must focus on the

- Triggering event
- Binding between event and workflow container
- Definition of employees responsible for workflow items

See also:

[Workflow Editor \[Page 86\]](#)

[Workflow Container \[Page 86\]](#)

[Triggering Events \[Page 86\]](#)

[Workflow Steps \[Page 87\]](#)

[Binding Between Workflow and Standard Task \[Page 88\]](#)

[Responsibilities \[Page 88\]](#)

[Exceptions \[Page 89\]](#)

If you require further information, see [Workflow Definition \[Ext.\]](#).

Workflow Editor

You can access the workflow editor from the task maintenance screen by choosing a task of type WS (standard workflow) or WF (customer workflow).

The system contains the following menu path: *Tools → SAP Business Workflow → Development → Definition tools → Tasks*.

There are two different editors:

1. Alphanumeric editor
2. Graphical editor

Workflow Container

To define a workflow, you must first define the workflow container. The container elements defined at this point are “global”, which means that the data they contain can be processed by any workflow step.

The document to be printed usually includes employer or employee data. For this reason, it is a good idea to define a container element of the **EMPLOYEE** type, or the country-specific **EMPLOYEEXX** version (XX = country ISO code).

Triggering Events

Each workflow has at least one triggering event that is exactly defined. The generation of events in HRMS is described in the [event generation \[Page 81\]](#) section.

If you do not set any event parameters, the event container includes a reference to the objects that generated the event. This value is stored in the **_EVT_OBJECT** standard container element, and is assigned to the **SWO_OBJECT** type.

Event data is stored in the *Container* event parameter. You must specify the binding between this container and the workflow container in the workflow definition.

Workflow Steps

The triggering event editor enables you to specify a separate binding for each triggering event.



In the above [Example: Standard Task for Printing a Contract \[Page 84\]](#), workflow is triggered by the *Social Insurance: Contract Changed* or *Social Insurance: Created event*. In accordance with the entries in table T588W, the events described at this point create an infotype 0061 record, or change a field (*contract type*) in the same infotype.

The container event parameter includes a standard reference to the object that triggered the event. This ensures that the information stored in infotype 0061 is complete.

The additional *basic pay* element is a help variable that makes it easier for you to access the *basic pay* wage type. The *document* element initializes the name of the document to be printed. This is merely an initial value because the real name of the document is determined on the basis of other workflow operations.

If you require further information, see [Event \[Ext.\]](#).

Workflow Steps

A workflow definition consists of a specific number of workflow steps. An activity that describes a task performed by an employee (see [Defining Standard Tasks \[Page 82\]](#)) represents a potential workflow item.

To enable you to define complex rules, you can use several types of step and control flow operation:

- **Condition, multiple condition**
- **Container operation**
- **User decision**
- **Event creator, wait step**
- **Loop**
- **Parallel process (fork)**
- **Block**
- **Process control**
- **Exit**

Each step is associated with an event that documents the change of status after the step has been performed.



If you require a container operation to set the document type, which means that the *document name* container element must be changed, the *document name set* event documents the change. In the example below, the result of the multiple condition is

Binding Between Workflow and Standard Task

represented by the *indefinido* (for Spain) or *other* events. Please note that you must insert additional branches at this point if there is more than one document.



These types of event are not “published” and only exist within workflow. If you want to create general, published events, use the *Event creator* step.

Binding Between Workflow and Standard Task

Use

After you have defined the printing rules, the document can be printed. A standard task that references an appropriate method of business object **STD_TEXT** has already been defined for this purpose (see [Example: Standard Task for Printing a Contract \[Page 84\]](#)). The standard task has its own container, which means you must define the binding between the workflow container and the standard task container.

The triggering event includes a reference to the *social insurance* object. This reference is written to the workflow container. In the next step, the information must be transferred to the standard task responsible for printing the document.

The standard task container also includes an element called *employee*. This information is required for printing employee-specific data (infotypes), and is also derived from the *social insurance* object type. Just like every other object type derived from infotype data, the object has an *aggregate* attribute that includes a reference to the corresponding **EMPLOYEE** object. The *aggregate* is also used to determine information on the personnel area (container element **PersonnelArea** in the standard task).

Responsibility

Use

The workflow definition includes the *Print document* step described in [Binding Between Workflow and Standard Task \[Page 88\]](#). This step must be processed by at least one employee (user), which means you must determine who is responsible for this task.

You can define responsibility for a task in accordance with the following criteria:

- Role
- Organizational unit
- Job
- Position

Exceptions

- Work center
- User
- Container element

These criteria are determined in Customizing and must be set up by the customer. To facilitate testing, it is a good idea to assign processing of this step to the *workflow initiator*. The *workflow initiator* is found in the workflow container under element `WF_INITIATOR`. This element includes the user name of the employee who triggered workflow.

The activity is only displayed in the office inbox of the appropriate employee if this step has been performed correctly.

If you require further information, see [Roles \[Ext.\]](#).

Exceptions

Use

If no errors occur, the standard task that you selected prints the event in accordance with the workflow definition. If an error occurs, this method gives rise to exceptions.

Possible exceptions must be processed within workflow. For this reason, you must specify an event name for each exception. Workflow creates a branch for each exception. A user decision is one possible reaction to an exception. This means that the user chooses to re-edit the document, or end workflow.

Using and Linking to Application Functions

[Create Documents using SAPscript \[Page 90\]](#)

[Define Main Text \[Page 90\]](#)

[Define Form \[Page 91\]](#)

[Use Container Element Data in Form Windows \[Page 92\]](#)

Create Documents Using SAPscript

Use

To create the document generated by workflow, you must use the R/3 word processing tools.

Activities

Perform the following steps:

- Create the main text.
- Create a form. This step is only required for complex layouts.

You can perform these steps by choosing

Tools → *Word processing* → *Standard text*

and

Tools → *Word processing* → *Form*

See also:

[Define Main Text \[Page 90\]](#)

[Define Form \[Page 91\]](#)

[Use Container Element Data in Form Windows \[Page 92\]](#)

Define Main Text

Use

The main text defines exactly **what** is printed. The form defines **how** the defined text is printed, that is, fonts, layouts, windows, page numbers, etc.



Printers are just one output device supported by SAPscript. Other devices include telefax, telex, office mail, and screen.

If you want to generate a letter or notification for internal use, you do not need to define or use a specific form. Every SAPscript document is assigned to a form. The default form is SYSTEM, which only contains a few fonts and formatting options.

You can assign a different form to a document by choosing *Format* → *Form*.

You can use text symbols in the text. A text between ampersands (that is, "&") is defined as a text symbol.

&Symbol&

User-defined symbols are set up in the SAPscript editor using the following commands:

Define Form

```
/: DEFINE &MY_SYMBOL& = 'Notification'
```

or

```
/: DEFINE &MY_SYMBOL& = &OTHER_SYMBOL&
```

When printing a text in workflow, you can also use container elements as symbols (see [Business Objects \[Page 80\]](#)). This facilitates the main task of automatic generation, that is, printing the contents of the database.

SAPscript also enables you to print lines, boxes, and shadowed boxes.

See also:

[Use Container Element Data in Form Windows \[Page 92\]](#)

Define Form

Use

As a rule, official documents sent to legal persons (regarding tax or social insurance, for example) have a fixed layout with well-defined print positions and dimensions. In such instances, you must define a form.

The form allows you to

- Define the width and height of windows.
- Position windows in a fixed area on the page.
- Assign windows to specific pages in the document.
- Assign text elements to specific windows.
- Define new paragraphs so you can use different fonts and font attributes.

The *MAIN* window is defined for every form. If a specific form is defined when a SAPscript text is output, the text is printed in the *MAIN* window. You must define the text printed in the window using text elements, which are valid for all documents assigned to the form.

To create forms based on A4 letter size, you can use the `S_TEMPLATE_DINA4` template.

Use Container Element Data in Form Windows

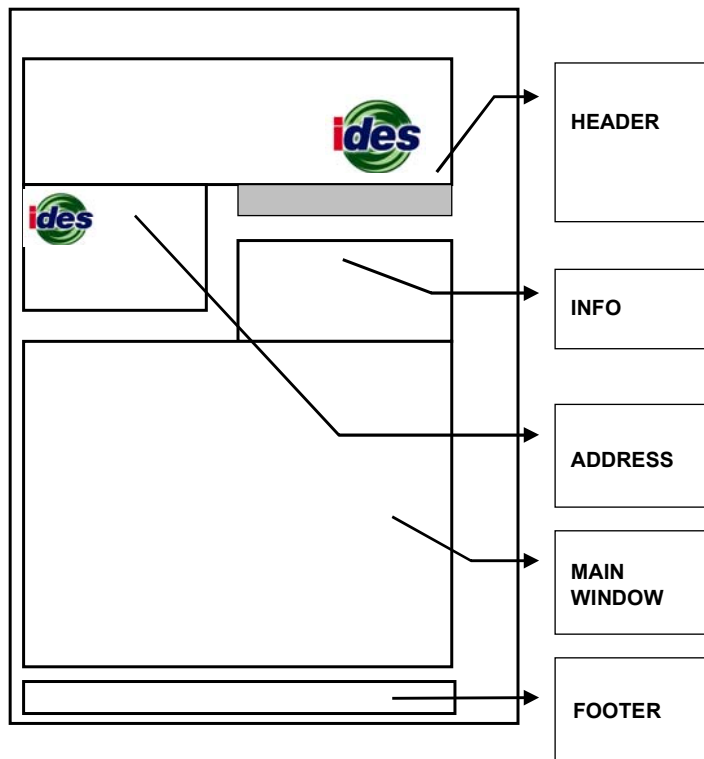


Figure: Form S_TEMPLATE_DINA4: Windows

See also:

[Use Container Element Data in Form Windows \[Page 92\]](#)

Use Container Element Data in Form Windows

In the main text, you can access employee data by using text symbols (`&Object.Attribute&`). However, you cannot use this method in form windows. One way to avoid this problem is by defining text-specific symbols using the **DEFINE** command.

- **Main Text**

```
/* „Define a “Help” symbol to create the employee address.
```

```
/: DEFINE &EE_ADDRESS&  
=&Employee.PersonnelData.StreetAndHouseNumber&
```

- **Form Window (Text Elements)**

```
/* Print employee address in a form window.
```

```
AS &EE_ADDRESS&
```

Printing a P46 Form (Company Cars) (PA-PAY-GB)

Text-specific symbols are recognized in the entire form, which means you can use the defined symbols in each window.

Printing a P46 Form (Company Cars) (PA-PAY-GB)

Purpose

The Inland Revenue must be notified if a company car is provided for the private use of an employee or a director. An employer is therefore required to complete a P46 form every quarter for employees who join the company car scheme for the first time, or employees that change cars during the tax year.

Process Flow

The workflow template WS00400251 (HR_GB: PrtCar) provided by SAP in the standard system enables you to print P46 forms whenever a situation arises where this is necessary.

The workflow scenario supports the following situations:

- The employee/director is provided with a company car that is available for private use
- A company car provided to the employee/director is replaced by another car that is available for private use
- The employee/director is provided with a second or further car that is available for private use
- The employee becomes a director

Technical Background (PA-PAY-GB)

Object Types

The interface between the R/3 functionality and SAP Business Workflow is implemented by the underlying object technology.

The following information is of a technical nature. You require this information if you are interested in the details regarding the implementation or you want to make your own modifications.

[Object Type BUSMILEAGE \(Business Mileage\) \[Page 94\]](#)

[Object Type CARCAPACIT \(Car Capacity\) \[Page 94\]](#)

[Object Type COMPCARGB \(Company Car GB\) \[Page 94\]](#)

[Object Type NINSURANCE \(National Insurance\) \[Page 95\]](#)

[Object Type PROLLCAR \(Payroll Car\) \[Page 95\]](#)

Standard Tasks

Standard tasks are single-step tasks supplied by SAP that describe elementary business activities from an organizational perspective.

[Standard Task TS00408216 \(HRGBBusLimit\) \[Page 95\]](#)

[Standard Task TS00408217 \(HRGBHubraum\) \[Page 96\]](#)

[Standard Task TS00408218 \(HRGBRegdate\) \[Page 97\]](#)

[Standard Task TS20000020 \(HRGBPrivPay\) \[Page 97\]](#)

[Standard Task TS20000042 \(HRGBConvert\) \[Page 98\]](#)

Workflow Template

The business process is implemented as a workflow definition within a workflow template. You can find this workflow template in your R/3 System:

[Workflow Template WS00400251 \(HR_GB:PrtCar\) \[Page 99\]](#)

Object Type BUSMILEAGE (Business Mileage)

The system reads table T5G61 containing the business mileage bands. This is a static operation without an output.

You can find the attributes and methods that have been defined in the object type definition in the Business Object Repository.

Position of the Object Type in the Business Object Repository

Personnel Management → Personnel Administration → Great Britain

Object Type CARCAPACIT (Car Capacity)

The system reads table T5G62 containing the car capacity bands. This is a static operation without an output.

You can find the attributes and methods that have been defined in the object type definition in the Business Object Repository.

Position of the Object Type in the Business Object Repository

Personnel Management → Personnel Administration → Great Britain

Object Type COMPCARGB (Company Car GB)

The system reads the *Company Cars* infotype for any changes to the information. This is a static operation without an output.

Object Type NINSURANCE (National Insurance)

You can find the attributes, methods, and events that have been defined in the object type definition in the Business Object Repository.

Position of the Object Type in the Business Object Repository

Personnel Management → Personnel Administration → EMPLOYEEET → contains

Object Type NINSURANCE (National Insurance)

The system reads the *National Insurance* infotype for any changes to the information. This is a static operation without an output.

You can find the attributes, methods, and events that have been defined in the object type definition in the Business Object Repository.

Position of the Object Type in the Business Object Repository

Personnel Management → Personnel Administration → EMPLOYEEI → contains

Object Type PROLLCAR (Payroll Car)

The system reads the payroll results. This is a static operation without an output.

You can find the attributes and methods that have been defined in the object type definition in the Business Object Repository.

Position of the Object Type in the Business Object Repository

Personnel Management → Personnel Administration → Great Britain

Standard Task TS00408216 (HRGBBusLimit)

This standard task determines the lower and upper limits of the business mileage bands.

Standard Task: 00408216

Abbreviation: HRGBBusLimit

Name: *Determine business mileage band*

Referenced Object Method, Attributes

Object Type: BUSMILEAGE

Method: FindMileageLimit

Attributes: *Background processing*

Agent Assignment

An agent assignment is not required since the standard task is processed in the background.

Standard Task TS00408217 (HRGBHubraum)

Other Information on the Standard Task

The object method operates on the work item by which it is referenced. The object processed is therefore the work item itself; this object is not created until runtime. The element **_WI_Object_Id** of the task container must be assigned a value.

The referenced method has one parameter (input), **BeginDateCompCar**, the date on which the company car was first made available to the employee.

For this method parameter, there is a corresponding element in the task container and a binding definition from the task container to the method parameter container.

The referenced method has two parameters which are the outcome of the calculations carried out by the standard task:

- **LowLimit** - Lower limit
- **UpLimit** - Upper limit

The output parameters allow the expected level of annual business mileage for the car to be placed in one of the three business mileage bands.

Standard Task TS00408217 (HRGBHubraum)

This standard task determines the lower and upper limits of the car capacity bands.

Standard Task: 00408217

Abbreviation: HRGBHubraum

Name: *Determine car capacity limit*

Referenced Object Method, Attributes

Object Type: CARCAPACIT

Method: FindCarCapacityLimit

Attributes: *Background processing*

Agent Assignment

An agent assignment is not required since the standard task is processed in the background.

Other Information on the Standard Task

The object method operates on the work item by which it is referenced. The object processed is therefore the work item itself; this object is not created until runtime. The element **_WI_Object_Id** of the task container must be assigned a value.

The referenced method has two parameters (input):

- **BeginDateCompCar** - Date the car was made available to the employee
- **EngineType** - Cylinder capacity of this car

For each method parameter, there is a corresponding element in the task container and a binding definition from the task container to the method parameter container.

The referenced method has two parameters which are the outcome of the calculations carried out by the standard task:

Standard Task TS00408218 (HRGBRegdate)

- **LowLimit** - Lower limit
- **UpLimit** - Upper limit

The output parameters allow the engine capacity of the car to be placed in one of the three bands.

Standard Task TS00408218 (HRGBRegdate)

This standard task determines the earliest date on which a company car was made available to the employee.

Standard Task: 00408218

Abbreviation: HRGBRegdate

Name: *Determine earliest available date*

Referenced Object Method, Attributes

Object Type: COMPCARGB

Method: FindEarliestUseOfCar

Attributes: *Background processing*

Agent Assignment

An agent is not required since the standard task is processed in the background.

Other Information on the Standard Task

The object method operates on the work item by which it is referenced. The object processed is therefore the work item itself; this object is not created until runtime. The element **_WI_Object_Id** of the task container must be assigned a value.

The referenced method has one parameter (input), **PersonnelNo**, the internal personnel number given to the employee.

For this method parameter, there is a corresponding element in the task container and a binding definition from the task container to the method parameter container.

The referenced method has one parameter which is the outcome of reading the table in the system. The output parameter states the earliest date the car was made available to the employee. This is the parameter **EarliestBegda**.

Standard Task TS20000020 (HRGBPrivPay)

This standard task calculates the private payments made by the employee for the private use of the car. This includes the cost of the fuel used for private motoring.

Standard Task: 20000020

Abbreviation: HRGBPrivPay

Name: *Determine private payment*

Standard Task TS20000042 (HRGBConvert)

Referenced Object Method, Attributes**Object Type:** PROLLCAR**Method:** FindPrivatePayment**Attributes:** *Background processing***Agent Assignment**

An agent is not required since the standard task is processed in the background.

Other Information on the Standard Task

The object method operates on the work item by which it is referenced. The object processed is therefore the work item itself; this object is not created until runtime. The element **_WI_Object_Id** of the task container must be assigned a value.

The referenced method has three parameters (input):

- **PersonalNo** - Internal personnel number given to the employee
- **CarInternalNo** - Internal car identifier number
- **PayrollArea** - Payroll area

For each method parameter, there is a corresponding element in the task container and a binding definition from the task container to the method parameter container.

The referenced method has two parameters which are the outcome of the calculations carried out by the standard task. The employee's contributions are split into the sum payable by the employee for the private use of the car and the amount paid for the fuel used for private motoring.

- **PrivatePayment** - Employee's contribution for private use of the car
- **FuelScaleCharge** - Fuel scale charge if employee is provided with fuel for private use of the car

Standard Task TS20000042 (HRGBConvert)

This standard task converts the data type DEC7 to the data type NUMC8. This is because DEC7 cannot be printed and must be converted to the numerical corrector to enable the P46 form to be printed.

Standard Task: 20000042**Abbreviation:** HRGBConvert**Name:** *Data type conversion DEC7 to NUMC8***Referenced Object Method, Attributes****Object Type:** COMPCARGB**Method:** ConversionBusinessMileage**Attributes:** *Background processing*

Workflow Template WS00400251 (HR_GB:PrtCar)**Agent Assignment**

An agent is not required since the standard task is processed in the background.

Other Information on the Standard Task

The object method operates on the work item by which it is referenced. The object processed is therefore the work item itself; this object is not created until runtime. The element **_WI_Object_Id** of the task container must be assigned a value.

The referenced method has three parameters (input):

- **OldMileageUpLimit** - Old mileage upper limit
- **OldMileageLowLimit** - Old mileage lower limit
- **OldBusinessMileage** - Km: Type: DEC7

For each method parameter, there is a corresponding element in the task container and a binding definition from the task container to the method parameter container.

The referenced method has three parameters which are the outcome of the conversion carried out by the standard task.

- **NewMileageUpLimit** - New mileage upper limit
- **NewMileageLowLimit** - New mileage lower limit
- **NewBusinessMileage** - Km: Type: NUMC8

Workflow Template WS00400251 (HR_GB:PrtCar)

This workflow template is used to print the P46 form for company cars once a company car has been made available to an employee, an employee has received another company car, a company car has been replaced, or an employee has been promoted to director. The connection to the workflow is achieved by a triggering event.

Workflow Template: 00400251

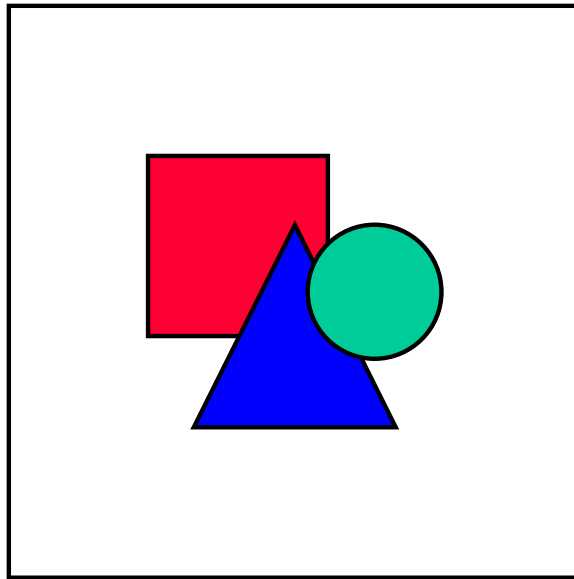
Abbreviation: HR_GB:PrtCar

Name: *Print company car form*

Triggering Events of the Workflow Template

Four events have been defined as the triggering events of the workflow template:

- Event **CREATED** of the object type **COMPCARGB**
- Event **FURTHERCAR** of the object type **COMPCARGB**
- Event **REPLACED** of the object type **COMPCARGB**
- Event **PROMOTED** of the object type **NINSURANCE**



The “linkage” between the event and the workflow template to be started is generally deactivated and must be activated if the workflow template is actually to be started. To do this, carry out the activity *Perform Task-Specific Customizing* under *SAP Business Workflow → Print Company Car Form* in *Customizing for Payroll: United Kingdom*.

Workflow Container

The respective elements must be available in the workflow container to hold the necessary information. The element **Company Car** was created in addition to the elements generally available to contain the object reference.

This item of information is available as an event parameter in the container of the triggering event and must be passed from there “per binding” to the workflow container.

Binding from the Event Parameter Container to the Workflow Container

The following binding is defined between the triggering events (event parameter container) and workflow container:

Workflow container		Event parameter container
Tax district	<-	_Evt_Object.Aggregate.PayrollAccountArea
Social insurance	<-	_Evt_Object.Aggregate.NationalInsurance
Personal data	<-	_Evt_Object.Aggregate.PersonalData
Employee	<-	_Evt_Object.Aggregate
Form P46	<-	HR_GB_Company_Car
Company Car	<-	_Evt_Object
Personnel area	<-	_Evt_Object.Aggregate.PersonnelArea
Event name	<-	_Evt_Name

Event Generation

Event Generation

Use

In the workflow template WS00400251 for printing a P46 form, the events involved are generated by calling a function module.

For this to occur, two relationships are defined in the system:

- Infotype table <-> business object type (T588V)
This is a simple link between an infotype and a business object
- Infotype table, operation <-> event, function module (T779W)
This enables events to be generated flexibly, depending on the operation performed on a particular infotype

In this workflow scenario, the function module HR_EVENT_RULES_PA0069 is responsible for generating events when particular changes are made to infotype 69 (*National Insurance*), while the function module HR_EVENT_RULES_PA0222 generates events for changes to infotype 0222 (*Company Cars*).

Maintenance of Event Receiver Linkage

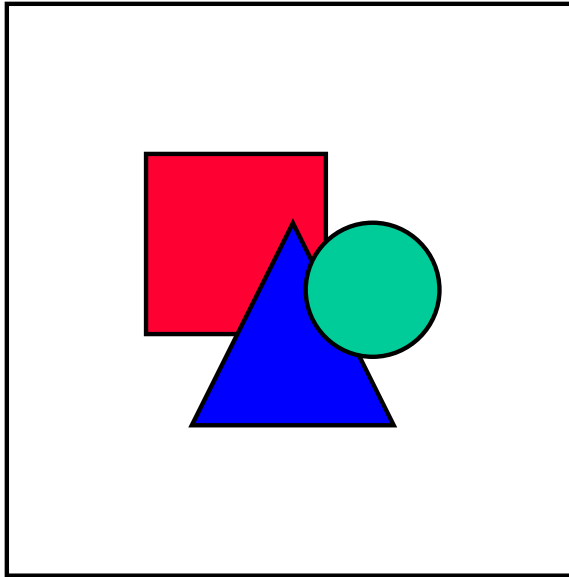
Use

So that the workflow template for printing the P46 form is actually started, you need to activate the linkage between the triggering events and the workflow template, which is the receiver of the events. You can do this in Customizing for *Payroll: United Kingdom*.

Activities

1. In Customizing for *Payroll: United Kingdom*, choose *SAP Business Workflow → Print Company Car Form → Perform Task-Specific Customizing*.
2. Execute the *Activate event linking* step.
3. In the *Event Linkage: Triggering Events* screen, expand the branch *Print company car form*. The four events used in the workflow scenario are displayed.
4. Activate the events.

Operation and Connection to Application Functionality (PA-PAY-GB)



You can also activate the event receiver linkage directly in the workflow template WS00400251 (*Print company car form*).

Operation and Connection to Application Functionality (PA-PAY-GB)

Purpose

The workflow WS00400251 (HR_GB:PrtCar) supports the process of completing and printing a P46 form whenever a situation arises where this is necessary.

Process Flow

When a user either creates a *Company Cars* record, changes the information in a *Company Cars* record, or changes particular information in a *National Insurance* record, this will start an event, as described below, triggering the workflow.

Entering Information in the *Company Cars* Infotype and Generating an Event

The action of creating (and saving) the *Company Cars* record creates a new object with the object type *CompCarGB* and automatically generates the event **created** for this object type. The event parameter container of this event contains the reference to the company car in the element **_Evt_Object**.

This event is one of the four *triggering events* of the workflow template for printing the P46 form (Company Cars).

Changing Information in the *Company Cars* Infotype and Generating an Event

1. The action of changing the *Regn. no.* and *From date* fields in the *Company Cars* record (and saving the changes) creates a new object with the object type *CompCarGB* and automatically generates the event **replaced** for this object type. The event parameter

Workflow Scenarios in Compensation Management (PA-CM)

container of this event contains the reference to the company car in the element **_Evt_Object**.

This event is one of the four *triggering events* of the workflow template for printing the P46 form (Company Cars).

2. The action of creating a new *Company Cars* record for a secondary car (and saving it) creates a new object with the object type *CompCarGB* and automatically generates the event **furthercar** for this object type. The event parameter container of this event contains the reference to the company car in the element **_Evt_Object**.

This event is one of the four *triggering events* of the workflow template for printing the P46 form (Company Cars).

Changing Information in the National Insurance Infotype and Generating an Event

The action of setting the *Company director* indicator (and saving the change) creates a new object with the object type *NInsurance* and automatically generates the event **promoted** for this object type. The event parameter container of this event contains the reference to National Insurance in the element **_Evt_Object**.

This event is one of the four *triggering events* of the workflow template for printing the P46 form (Company Cars).

Printing the P46 Form (Company Cars)

The user, ie. the workflow initiator, will find a workflow item in his or her integrated inbox. When this work item is executed, the P46 form (Company Cars) can be printed.

Workflow Scenarios in Compensation Management (PA-CM)

Definition

Compensation Management Workflow includes the following workflow templates:

[Compensation Adjustment Approval Process \(CMPApprProc\) \[Page 103\]](#)

[Follow-up Actions After Award Exercised \(CMPEexerc\) \[Page 107\]](#)

[Award Expiration \(CMPEexpire\) \[Page 109\]](#)

Compensation Adjustment Approval Process (CMPApprProc)

Definition

You can use this workflow template to approve (and therefore activate) compensation adjustments after the adjustments' approval has been requested.

Note that the adjustments, such as salary increases and bonuses, are granted in the *Compensation Management* application component, not in workflow. The connection to workflow

Compensation Adjustment Approval Process (CMPApprProc)

is facilitated by a triggering event, which in this case is the request for compensation adjustments.



If you change statuses without using workflow (that is, by using the application), you cannot continue to use workflow to change statuses or activate adjustments. In this case, you must use compensation administration to continue the approval process.

Workflow template: WS1000083

Abbreviation: CMPApprProc

Description: compensation adjustment approval process

Triggering Event of Workflow Template

The `created` event (work queue: compensation created) of the BUS7022 object is the triggering event for this workflow template.



In the R/3 System, "linkage" between the event and the workflow template to be started is generally deactivated. If you want to start the workflow template, you must activate linkage when you configure your system.

Binding From Event Parameter Container to Workflow Container

The following binding is defined between the triggering event (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
_WF_Initiator	←	&_EVT_CREATOR&
Work queue	←	&_EVT_OBJECT&



This workflow template uses work queues that are written to the database. To prevent the database from becoming too full (which could impede performance), delete completed work queues at regular intervals. Completed work queues are flagged as *completed*.

See also:

[Approve and Activate Compensation Adjustments \[Page 105\]](#)

Approve and Activate Adjustments

Approve and Activate Adjustments

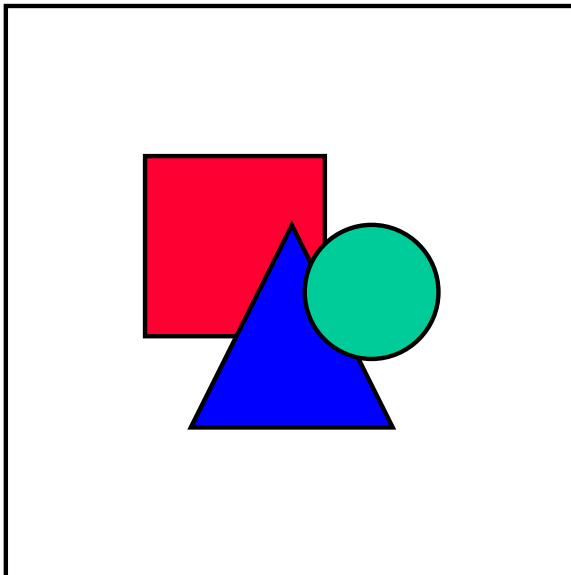
Purpose

The following is a possible process for approving and activating adjustments, for example, salary increases, bonuses, using workflow.

Process Flow

1. The department head runs the compensation administration process for the employees for which he/she is responsible and awards them adjustments.
The adjustments have the status *planned*.
2. The department head is happy with the adjustments and submits them to his/her manager.
The adjustments have the status *submitted*, this action **starts** a workflow.

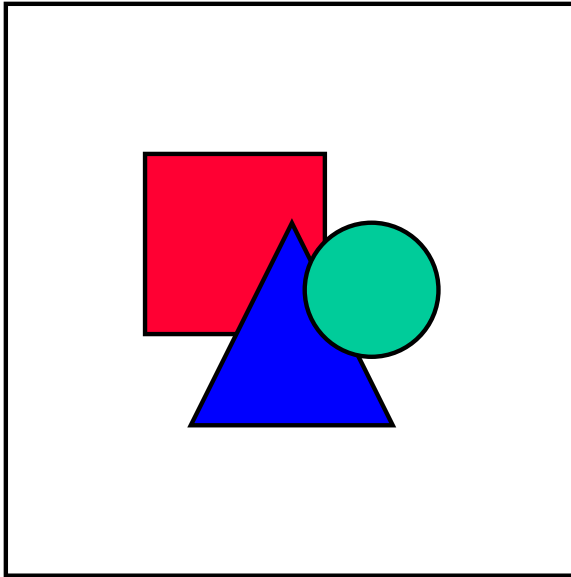
In technical terms, changing the status to *submitted*, triggers the created event for object BUS7022 and the workflow template 1000083 is started. The system uses the organizational plan to determine who is the manager of the department head.
3. The manager receives a work item in his/her integrated inbox and can either approve or reject the adjustment proposals.
4. If the manager approves the adjustments, then the system changes the status of the adjustments to *approved*.



At this point of the workflow, you can add an additional approval step, so that the manager's manager has to approve the adjustments before they can be activated.

5. If the manager has approved the adjustment proposals, the personnel administrator assigned to these employees can now activate the adjustments and the workflow ends here. The relevant personnel administrator receives the work item.
6. If the manager rejects the adjustments, the status is changed to *planned*

Approve and Activate Adjustments



You cannot just reject the adjustment for one employee, you have to reject all employees. The manager can create a note detailing the reasons for rejecting the adjustments at this stage.

7. The head of department that submitted the adjustments is informed and can then decide whether to withdraw the adjustment proposal or to rework the adjustments and resubmit them.

If the head of department decides to withdraw his/her proposal, then the workflow ends and the adjustments remain in the status *planned*.

If the head of department decides to cancel, the work item remains in his/her integrated inbox to be resubmitted.
8. The head of department decides, however, to rework the adjustments in accordance with the manager's suggestions and resubmits them.
9. The manager receives a work item in his/her inbox informing him/her that the proposals have been resubmitted and await his/her approval.
10. The manager approves the adjustments, the system changes the status of the adjustments to *approved*.
11. The adjustments can now be activated
12. The workflow ends here.

Result

The adjustments are approved and activated. The system creates the relevant infotype records for the employees.

Maintenance of Event Receiver Linkage

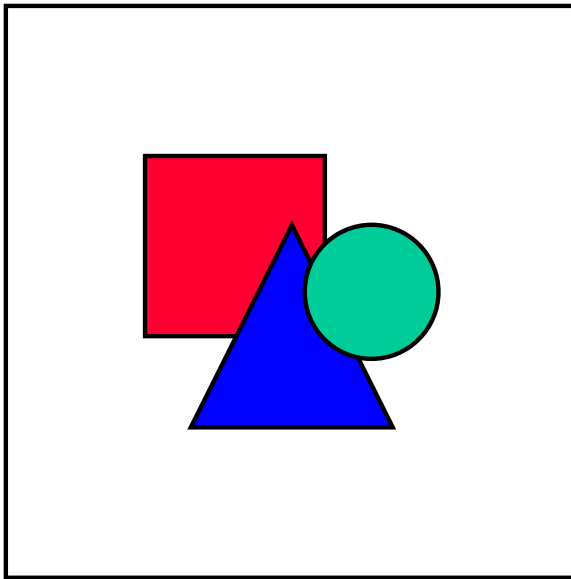
Maintenance of Event Receiver Linkage

Use

So that the workflow template for approving compensation adjustments is actually started, you need to activate the linkage between the triggering event and the workflow template, which is the receiver of the events. You can do this in Customizing for *SAP Business Workflow*.

Activities

1. In Customizing, choose *Basis Components* → *Business Management* → *SAP Business Workflow* → *Perform task-specific Customizing*.
2. In the application component hierarchy displayed, choose *Personnel Management* → *Compensation Management*, and then choose *Activate event linking*.
3. In the *Event Linkage: Triggering Events* screen for Compensation Management, choose *Approval Process Compensation Adjustments*
The events used in this workflow scenario are displayed.
4. Activate the events.



You can also activate the event receiver linkage directly in the workflow template WS100083 (Approval Process Compensation Management) .

Follow-up Actions After Award Exercised (CMPEexerc)

Definition

You use this workflow template to complete all the administrative steps necessary, once the employee has exercised his or her awards.

Workflow template: 1200169

Abbreviation: CMPEexerc

Description: Follow-up Actions After Award Exercised

Triggering Event of Workflow Template

The event `CreatedVisESS` (Exercise generated), for the object EMPAWEXERC (Award exercising) is the triggering event for this workflow template.



In the SAP R/3 System, the "linkage" between the event and the workflow template to be started is generally deactivated and you must activate it when you configure your system if you want to start the workflow template.

Binding from the Event Parameter Container to the Workflow Container

The following binding between the triggering event (event parameter container) and the workflow container is defined:

Workflow Container		Event Parameter Container
Award	←	&EVT_OBJECT.AGGREGATE&
Exercising	←	&_EVT_OBJECT&

See also:

[Exercising Awards \[Page 108\]](#)

Exercising Awards

Purpose

This process enables awards to be exercised with the help of Workflow.

Prerequisites

- You have maintained the event receiver linkage for the workflow template *Follow-up Action After Award Exercised* (CMPEexerc).



For more information, see the *Compensation Management IMG* under *Planning and Administration* → *Long-Term Incentives* → *Workflow* → *Set up 'Follow-up Action after Award Exercised' Workflow*.

- The relevant employee has been granted *incentive stock options* and *nonqualified stock options*, which have been awarded.
- You are using the relevant ESS service to exercise awards.

Award Expiration (CMPEXpire)

Process Flow

1. The employee exercises the award using the ESS service *Exercises Awards* in the company Intranet.
2. The system sends a work item to the HR administrator responsible for this employee.
3. The HR administrator receives the work item, which contains information on the award which has been exercised.
4. The HR administrator orders authorizes the bank
to buy a certain number of shares and to credit them to the employee's securities account.
to transfer a certain number of shares from the company's securities account to the
employee's securities account.
5. The bank carries out the request and informs the HR administrator of the exact price
at which the shares were bought.
at the time of the transfer.
6. The HR administrator enters the price as the fair market price in the work item and confirms
that the he or she has carried out the relevant follow-up actions.



The fair market value is required so that the employee may be taxed correctly.

7. The system deletes the work item from the HR administrators integrated inbox.

Result

The award has been exercised.

Award Expiration (CMPEXpire)

Definition

This workflow template enables you to ensure that your employees receive an automatic reminder two months before their awards expire.

Workflow template: 1200178

Abbreviation: CMPEXpire

Description: award expiration

Triggering Events of Workflow Template

The `ExpirationDateChange` (all options exercised) and `Created` (object created) events of the EMPAWARD object are triggering events for this workflow template.

Notification of Expiration of Awards



In the R/3 System, "linkage" between the event and the workflow template to be started is generally deactivated. If you want to start the workflow template, you must activate linkage when you configure your system.

Binding From Event Parameter Container to Workflow Container

The following binding is defined between the triggering events (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
_WF_INITIATOR	←	&EVT_CREATOR&
Award	←	&EVT_OBJECT&



This workflow can take a very long time to complete, at most from the date on which the awards are granted to the date on which they expire. In other words, the status can be set to *started* for several years.

See also:

[Notification of Expiration of Awards \[Page 110\]](#)

Notification of Expiration of Awards

Purpose

This process enables you to notify employees of the imminent expiration of their awards with the help of Workflow.

Prerequisites

- You have maintained the event receiver linkage for the workflow template Expiration of Awards (CMPEXpire).



For more information see the *Compensation Management* IMG under *Administration* → *Awards* → *Workflow* → *Set up Award Expiration Workflow*.

- Awards which are granted to employees have a fixed expiration date.

Process Flow

- One of the following events occurs:

The line manager or the compensation manager grants an employee an award.

Workflow Scenarios in Personnel Development (PA-PD)

The expiration date of an award that has already been granted changes due to a life event.

2. The system checks whether the award has a definite expiration date.
3. If so, the system sends the employee notification of the expiration of the award two months in advance.

Result

The employee is informed of the imminent expiration of the award in due time and can decide whether he or she wishes to exercise the award in the time remaining .

Workflow Scenarios in Personnel Development (PA-PD)

Qualification Monitoring (PA-PD)

Purpose

This workflow describes how qualifications with validities are monitored.

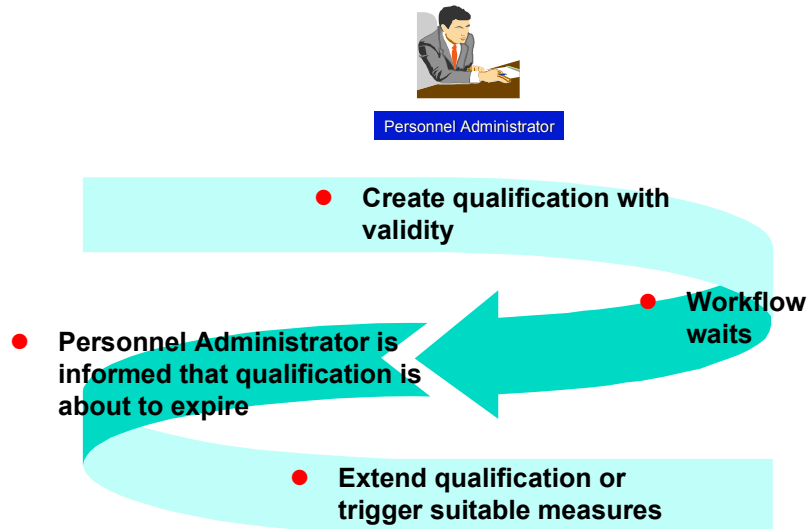
You use this workflow to inform the appropriate administrator about qualifications that are about to expire. This means that the administrator in question can extend the validity of the qualification, or trigger a personnel development measure (book the holder of the qualification onto a training course, for example).



This workflow only includes qualifications for which a [validity \[Ext.\]](#) was defined in the qualifications catalog.

Process Flow

Qualification Monitoring



1. A personnel administrator creates a qualification for an employee, and defines a validity for this qualification. The workflow is triggered when the qualification is created.
2. The end date of the validity is determined on the basis of the start date of the qualification and the duration of the qualification's validity. The workflow waits in a queue.
3. The administrator is informed that the qualification is about to expire per work item. The administrator is informed of this at a predefined point in time before the end date of the qualification. The fact that the qualification is about to expire is determined by means of a profile matchup.



If the qualification is refreshed (or its validity extended) before the work item is executed, the existing workflow is terminated, and a new workflow is started.

4. The workflow ends when the work item is executed.

Result

The personnel administrator can now take the measures required.

Technical Implementation (PA-PD)

Technical Implementation (PA-PD)

Object Types

Object-based technology means that it is possible to create an interface between R/3 functionality and the Workflow System.

The information that follows is very technical. You only need to read it if you require detailed technical information on implementing workflow, or if you want to make enhancements of your own.

[Object Type BUS7019 \(PA-PD\) \[Page 113\]](#)

Standard Tasks

The standard tasks that come with the standard SAP System (one-step tasks) describe elementary business activities from an organizational point of view.

The [standard task PDProfileMat \(PA-PD\) \[Page 114\]](#), which is used here, is available in the form of a module, and can be used as it is (i.e. without being changed) in other workflow scenarios.

Roles

The role entitled [Role of Personnel Administrator \(PA-PD\) \[Page 114\]](#) is used to determine the agent.

Workflow Templates

The business process flow has been provided as a workflow template. You will find the [workflow template PDExpiredQua \(PA-PD\) \[Page 115\]](#) in your R/3 System.

Object Type BUS7019 (PA-PD)

Definition

A qualification held by a person is a business application object of type **BUS7019**.

To view technical data on the object type, go to the Business Object Repository (*Personnel management* → *Personnel development*).

Use

In the scenario, the validity of an object of this type is evaluated.

Structure

Key Fields

An object of this type is identified uniquely by means of the following fields:

- PlanVersion

Standard Task PDProfileMat (PA-PD)

- PersonType
- PersonID
- QualificationID
- EndDate

Standard Task PDProfileMat (PA-PD)

Definition

When this standard task is executed, a profile matchup is carried out, and the result of this matchup (a list identifying the qualification that is about to expire) is passed on to the personnel administrator in the form of a work item.

Standard task: 1000136

Abbreviation: PDProfileMat

Name: *Profile Matchup*

Structure

Method Referenced, Properties

Object type: BUS7019 (*Qualification*)

Method: RHPPSHOWPROFILEMATCHUP (*Profile Matchup*)

Properties: *Confirm end of processing*

Agent Assignment

At runtime, this standard task is addressed to the employee's personnel administrator using role resolution. Since the employee in question can be any employee, the task must be classified as a *general task*.

Role of Personnel Administrator (PA-PD)

Definition

Role used for determining the personnel administrator responsible for an employee.

Role: 01000012

Abbreviation: pdPosPersAdm

Workflow Template PDExpiredQua (PA-PD)

Name: *Personnel Administrator*

Structure

Role Parameters

The following role parameters are passed on:

- **ORG_USER** (reference field **RHOBJECTS-OBJECT**); contains the concatenated keys (type and ID) of organizational objects
- **PlanVersion** (reference field **HRSOBID-PLVAR**); contains the 2-character ID of a plan version

The role parameters exist as elements in the role parameter container.

Binding Definition

The role is used to find the correct personnel administrator for the employee (starting out from the initiator of the workflow). The position of the workflow initiator is determined, and, when the work item is executed, the current holder of this position.

The user name of the initiator is stored in a 14-character field (reference **RHOBJECTS-OBJECT**) in the Element **_WF_Initiator** of the workflow container. This means that the binding is defined in the corresponding step for the element **ORG_USER**.

Integration

Preparations in the Organizational Plan

The position of the workflow initiator is determined for the purposes of role resolution. If the role is to be resolved correctly, each possible agent must hold only one position.

Workflow Template PDExpiredQua (PA-PD)

Definition

Workflow template for determining qualifications that are about to expire.

Workflow template: 1000089

Abbreviation: PDExpiredQua

Name: Expired Qualification

Structure

Workflow Container

The following information must be available during the workflow:

- the *(object) reference to the qualification that is about to expire*

Workflow Template PDExpiredQua (PA-PD)

- the *name of the agent*

In order to process this information, the following elements must be available in the workflow container:

- The element **_WF_Initiator** for processing the agent is already defined in the workflow container in the standard system.
- The element **Qualification** for processing the object reference was added to the existing standard system elements.

The information is available in the form of event parameters in the container of the triggering event, and has to be passed on from there using a binding to the workflow container.

Triggering Event of the Workflow Template

The event **CREATEDWITHVALIDITY** (*Qualification created with validity*) is specified for object type **BUS7019** (*Qualification held by a person*) as the triggering event of the workflow template. This "link" between the event and the workflow template to be started has to be activated first in Customizing for this workflow template.

Binding of Event Parameters to the Workflow Container

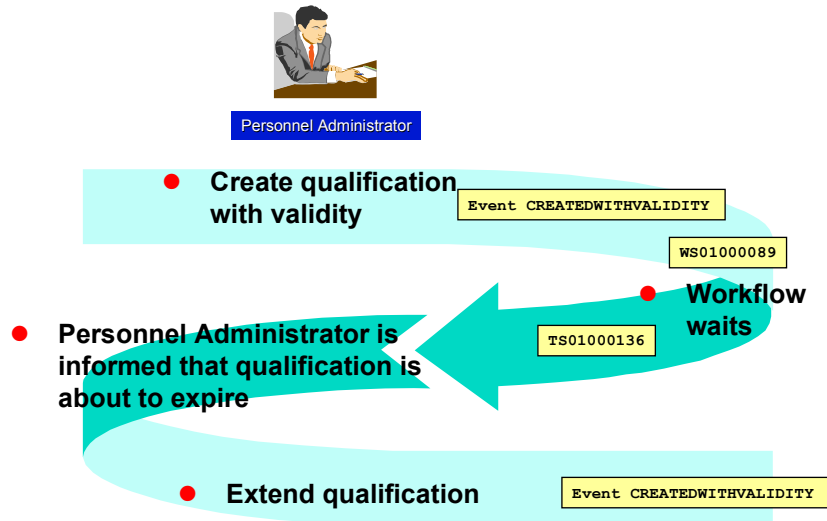
The following binding is defined between the triggering event (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
_WF_Initiator	<-	_Evt_Creator
Qualification	<-	_Evt_Object

Preparations and Customizing (PA-PD)

Process Flow

Qualification Monitoring



Preparations and Customizing (PA-PD)

Once you have carried out general customizing, which ensures that the Workflow System functions correctly, you need to customize the workflow template itself.

Prerequisites

You have completed general Customizing for the SAP Business Workflow system.

Setting Up the Organizational Plan

The qualification is created by a personnel administrator, and must be extended or renewed by this person (or his/her successor). The personnel administrator must hold a position. See [Organizational Plan \[Ext.\]](#).

Working Through Task-Specific Customizing

In **Customizing** for the SAP Business Workflow system, the standard task 1000136 (*Profile Matchup*) has to be assigned to possible agents.

Procedure

1. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
2. Choose the step *Assign task to agent* in the *Personnel Development* component.
3. Classify standard task 1000136 as a general task.

Linking Events to Event Receivers

The event **CREATEDWITHVALIDITY** defined for object type **BUS7019** is the triggering event for the workflow template **1000089** and, consequently, is specified in the linkage table in the standard system.

To ensure that the workflow template really is started, you must **activate** the link between the triggering event and the workflow template as the receiver of the event. You do this in **Customizing** for SAP Business Workflow.

Procedure

1. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
2. Activate the link for the workflow template *Appraisal* in the *Personnel Development* application component.

Alternatively, you can activate the link between the event and the event receiver by editing the workflow template **PdvAppraisal** directly.

Connection to Application Functions (PA-PD)

Prerequisites

You have made the settings described in the topic [Preparations and Customizing \(PA-PD\) \[Page 117\]](#).

Process Flow

A personnel administrator creates a qualification for an employee, and defines a validity for this qualification. The personal administrator enters this information in the employee's *Qualifications* subprofile.

Approving and Following Up an Appraisal (PA-PD)

This generates the event **CREATEDWITHVALIDITY** (*Qualification created with validity*), which triggers the workflow. See [Workflow Template PDExpiredQua \(PA-PD\) \[Page 115\]](#). The workflow comprises two steps that run parallel to each other: Wait for Renewal, and Matchup Completed.

Matchup carried out

This step sends a work item to the personnel administrator at a specific point in time, and informs him or her that the qualification is about to expire. A profile matchup is carried out to provide information on the qualification.

The start date is the end date of the qualification's validity period minus a period defined in the appraisal model.

The personnel administrator can now decide what to do next.

Wait for renewal

If the qualification is renewed before the work item is sent, the workflow is terminated automatically.

Remainder of Process Flow

When the qualification is renewed, a new workflow is triggered.

Approving and Following Up an Appraisal (PA-PD)

Purpose

This workflow describes the procedure for approving and following up an appraisal.

Once an appraiser has held and completed an appraisal, it can be sent to the appraisee for approval. The appraisee can display the appraisal, and opt to approve or reject it. If the appraisee decides to reject the appraisal, the appraiser can then review the appraisal. Once the appraisal has been approved, various follow-up activities can be performed.



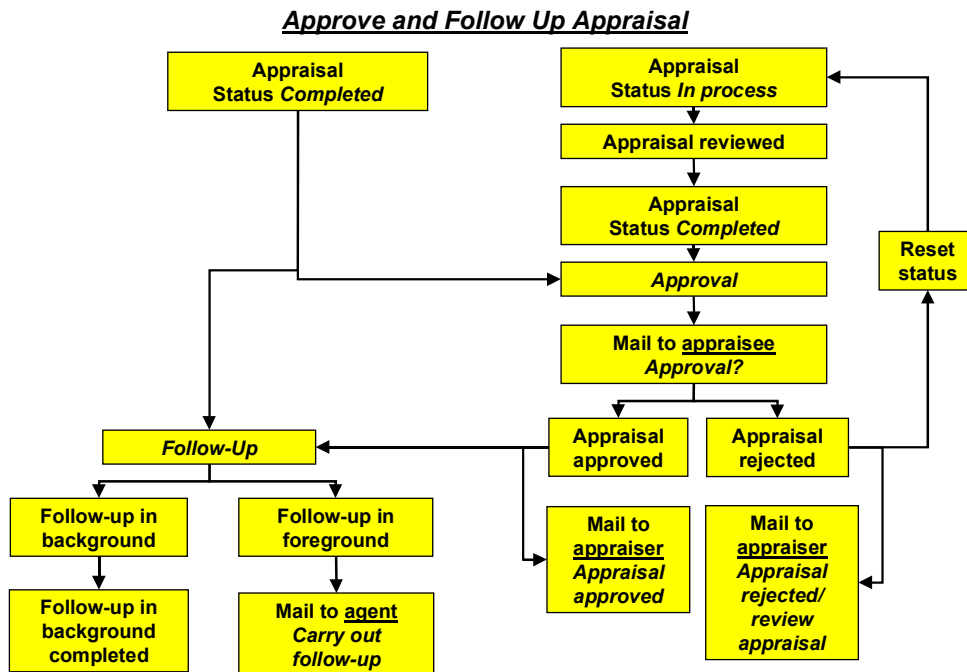
You must use this workflow if you wish to perform the follow-up activities defined for the appraisal model. You can also set up the workflow so that it omits the approval process.

You use this workflow if, for example, you want to:

- Give appraisees the option of approving/rejecting an appraisal
- Give yourself the option of reviewing rejected appraisals
- Update specific qualifications profiles on the basis of the results of appraisals
- Perform remuneration reviews in conjunction with appraisals
- Make appraisal results available to other departments for follow-up/further processing
- Make the following up of an appraisal depend on whether it has been approved or not

Approving and Following Up an Appraisal (PA-PD)

Process Flow



1. An appraiser carries out an appraisal, and completes this appraisal. The completion of the appraisal is the triggering event that starts the workflow.
2. It is the appraisee's task to approve the appraisal. This person is determined automatically on the basis of the appraisal data defined in the system.
3. The appraisee receives the appraisal. He or she can display it, and opt to approve or reject it.

Appraisal approved	Appraisal rejected
---------------------------	---------------------------

Technical Implementation (PA-PD)

<p>a. The appraisal is informed of this by e-mail.</p>	<p>a. The status of the appraisal is changed back to <i>In process</i> so that the appraisal can be reviewed and changed if necessary.</p> <p>b. The appraiser is informed by e-mail that the appraisal was rejected – the appraisal itself is attached to this mail so that it can be reviewed.</p> <p>c. Once the appraisal has been reviewed and changed as necessary, it is completed and passed on to the appraisee again for approval.</p> <div data-bbox="954 661 1003 709" data-label="Image"> </div> <p>An appraisal can be rejected several times. This step is not complete until the appraisal has been approved.</p>
--	---

4. Once the appraisal has been approved, it is followed up.

Follow up in background	Follow up in foreground
This runs automatically, and cannot be canceled. No correspondence is triggered.	In this case, a suitable agent is determined – this agent is informed of this task by e-mail. The agent can carry out or cancel the follow-up activities.

Result

The workflow is completed.

- The status of the appraisal is *Approved* or *Completed*.
- The background follow-up activities were performed automatically.
- The foreground follow-up activities were performed as required.

For more information on defining the workflow, refer to [Details of Workflow Definition \[Page 130\]](#).

Technical Implementation (PA-PD)**Object Types**

Object-based technology means that it is possible to create an interface between R/3 functionality and the Workflow System.

The information that follows is very technical. You only need to read it if you require detailed technical information on implementing workflow, or if you want to make enhancements of your own.

[Object Type BUS7026 Appraisal \(PA-PD\) \[Page 122\]](#)

Standard Tasks

The standard tasks that come with the standard SAP System (one-step tasks) describe elementary business activities from an organizational point of view.

The standard tasks that are used here, is available in the form of modules, and can be used as they are (i.e. without being changed) in other workflow scenarios.

- [Standard Task 01000160 PdvApprove \[Page 123\]](#)
- [Standard Task 01000161 PdvReject \[Page 124\]](#)
- [Standard Task 01000169 PdvApprDecis \[Page 124\]](#)
- [Standard Task 01200194 PdvStatus \[Page 125\]](#)
- [Standard Task 01200195 PdvChangAppr \[Page 125\]](#)
- [Standard Task 01200201 PdvAssesBack \[Page 126\]](#)
- [Standard Task 01200202 PdvAssesDia \[Page 127\]](#)

Roles

Roles are used to determine the appraiser and appraisees.

- [Role 01000016 Appraiser \[Page 127\]](#)
- [Role 01200025 Supervisor of Appraisee \[Page 128\]](#)
- [Role 01000017 Appraisee \[Page 129\]](#)

Workflow Templates

The business process flow has been provided as a workflow template. You will find the following workflow template in your R/3 System:

[Workflow Template 01000105 PdvAppraisal \[Page 129\]](#)

The following workflow template is defined as a subworkflow in the workflow template *PdvAppraisal* (see [Workflow Definition Details \[Page 130\]](#)):

- [Workflow Template 01000108 PdvAppEmpl \[Page 133\]](#)

Object Type Appraisal (PA-PD)

Definition

An appraisal is a business application object of type **BUS7026**.

To view technical data on the object type, go to the Business Object Repository, or “BOR” (*Personnel management* → *Personnel development*).

Standard Task PdvApprove (PA-PD)**Use**

When you create an appraisal, an object of this type is generated. The appraisal process comprises several phases. During each of these phases, the appraisal itself has a specific status.

When an appraisal is completed, it is assigned the status *Completed*, and a historical record is created for it. At the same time, the event **Appraisal.CompletedAppraisal** (Complete appraisal) is triggered.

In workflow template 1000105 ([Appraisal \[Page 119\]](#)), a completed appraisal is approved, rejected, or followed up.

Structure**Key Fields**

An object of this type is identified uniquely by means of the *Plan version* and *AppraisalID* fields.

See also:

[Appraisals \[Ext.\]](#)

Standard Task PdvApprove (PA-PD)**Definition**

When this standard task is executed, the status of an appraisal is changed to *Approved*.

Standard task: 1000160

Abbreviation: PdvApprove

Name: Approve Appraisal

Structure**Method Referenced, Properties**

Object type: BUS7026 (*Appraisal*)

Method: RHPAAPPRAISALAPPROVE (*Appraisal - Approve*)

Properties: Background processing

Agent Assignment

There is no agent assignment since the standard task is processed in the background.

Standard Task PdvReject (PA-PD)

Definition

When this standard task is executed, the status of an appraisal is changed to *Rejected*.

Standard task: 1000161

Abbreviation: PdvReject

Name: *Reject Appraisal*

Structure

Method Referenced, Properties

Object type: BUS7026 (*Appraisal*)

Method: RHPAAPPRAISALREJECT (*Appraisal - Reject*)

Properties: *Background processing*

Agent Assignment

There is no agent assignment since the standard task is processed in the background.

Standard Task PdvApprDecis (PA-PD)

Definition

When this standard task is executed, the appraisal is displayed, together with a decision task. The appraisal can be approved or rejected.

Standard task: 1000169

Abbreviation: PdvApprDecis

Name: *Approve decision*

Structure

Method Referenced, Properties

Object type: DECISION

Method: PROCESS

Properties: *Internet enabled*

Standard Task PdvStatus (PA-PD)

Agent Assignment

At runtime, this standard task is addressed to the [appraisee \[Page 129\]](#) using role resolution. Since all the objects that can be appraisees are possible agents of the task, you must classify the task as a *general task*.

Alternatively, the role [Supervisor of Appraisee \[Page 128\]](#) can also be included. This makes sense, for example, if forms of appraisal are used where the objects appraised cannot carry out any actions on their own (business events, for example).

Standard Task PdvStatus (PA-PD)

Definition

When this standard task is executed, the status of an appraisal is changed back (reset) to *In process*.

Standard task: 1200194

Abbreviation: PdvStatus

Name: *Reset Status of Appraisal*

Structure

Method Referenced, Properties

Object type: BUS7026 (*Appraisal*)

Method: APPRAISALTAKBACK (*Appraisal – Reset Status to Active*)

Properties: *Background processing*

Agent Assignment

There is no agent assignment since the standard task is processed in the background.

Standard Task PdvChangAppr (PA-PD)

Definition

When this standard task is executed, the appraisal is called up in Change mode. The appraisal can be reviewed and completed again. This starts the approval process again.

Standard task: 1200195

Abbreviation: PdvChangAppr

Name: *Change and Complete Appraisal*

Structure

Method Referenced, Properties

Object type: BUS7026 (*Appraisal*)

Method: EDIT (*Change Object (Appraisal)*)

Agent Assignment

At runtime, this standard task is addressed to the [appraiser \[Page 127\]](#) using role resolution. Since all the objects that can be appraisers are possible agents of the task, you must classify the task as a *general task*.

Alternatively, the role [Supervisor of Appraisee \[Page 128\]](#) can also be included. This makes sense if, for example, you use appraisal models where the appraiser cannot necessarily be identified (appraisal types *Multisource appraisal*, *Anonymous multisource appraisal*, and *Anonymous individual appraisal*).

Standard Task PdvAssesBack (PA-PD)

Definition

When this standard task is executed, follow-up activities are performed for the appraisal in the background.

Standard task: 1200201

Abbreviation: PDVAssesBack

Name: *Follow-Up Processing of an Appraisal in the Background*

Structure

Method Referenced, Properties

Object type: BUS7026 (*Appraisal*)

Method: SHOWPROCESSINGBACKGROUND (*Execute Further Processing Modules*)

Properties: *Background processing*

Agent Assignment

There is no agent assignment since the standard task is processed in the background.

Standard Task PdvAssesDia (PA-PD)

Standard Task PdvAssesDia (PA-PD)

Definition

When this standard task is executed, follow-up activities are performed for the appraisal in the foreground.

Standard task: 1200202

Abbreviation: PDVASSESDIA

Name: *Follow-Up Processing of an Appraisal in the Foreground*

Structure

Method Referenced, Properties

Object type: BUS7026 (*Appraisal*)

Method: SHOWPROCESSINGDIALOGUE (*Execute Further Processing Modules*)

Agent Assignment

At runtime, this standard task is addressed to a suitable agent using role resolution (see the role of [Supervisor of Appraisee \[Page 128\]](#)). The task must be classified as a *general task*.

Role of Appraiser (PA-PD)

Definition

Role used to determined the appraiser of an appraisal.

Role: 01000017

Abbreviation: PdvAppraiser

Name: *Appraiser*

Structure

Role Parameters

The role parameter that is passed on is **APPRAISAL**; the following fields are also passed on:

- **PlanVersion** - contains the 2-character ID of a plan version
- **AppraisalID** - contains the 8-digit numeric ID of an appraisal

Binding Definition

The role is used to find the appraiser of an appraisal (that is, the person who created and held the appraisal). The *ID* and *plan version* of the appraisal are stored in the **APPRAISAL** element of the workflow container. This means that the binding is defined in the corresponding step for this element.

Role of Supervisor of Appraisee (PA-PD)

Definition

Role used to determine the agent responsible for processing an appraisal.

Role: 1200025

Abbreviation: pdvAPPManage

Name: *Supervisor of Appraisee*

Structure

Role Parameters

The role parameter that is passed on is **APPRAISAL**; the following reference field is also passed on:

- **OBJECT** contains the 14-character key (derived from the object type and ID) of an object

Binding Definition

The role is used to find the appropriate processor for an appraisal.

The following options are available:

- The object appraised is of the object type *Business event* or *Business event type* – in this case, the course instructor is determined.
- The appraisal is based on an appraisal model with the appraisal type *Multisource appraisal*, *Anonymous multisource appraisal*, or *Anonymous individual appraisal* – in this case, the supervisor of the appraisee is determined. This is done by identifying the position of the appraisee and finding the holder of the corresponding chief position.
- Other cases – the appraiser is determined.

The object type and ID are stored in the **APPRAISAL** element of the workflow container. This means that the binding is defined in the corresponding step for this element.



If the appraisal models in the approval process could give rise to the situations listed above, use this role in tasks **TS 1200195** [Change and Complete Appraisal \[Page 125\]](#) and **TS 1000169** [Approve Decision \[Page 124\]](#) instead of the role [Appraiser \[Page 127\]](#).

Integration

Preparations in the Organizational Plan

Role of Appraiser (PA-PD)

The chief position for the appraiser is determined for the purposes of role resolution. If the role is to be resolved correctly, each possible agent must hold only one position.

Role of Appraiser (PA-PD)

Definition

Role used to determine the appraiser for an appraisal.

Role: 01000016

Abbreviation: PdvAppraiser

Name: Appraiser

Structure

Role Parameters

The role parameter that is passed on is **APPRAISAL**; the following fields are also passed on:

- **PlanVersion** - contains the 2-character ID of a plan version
- **AppraisalID** - contains the 8-digit numeric ID of an appraisal

Binding Definition

The role is used to find the appraiser for an appraisal. The *ID* and *plan version* of the appraisal are stored in the **APPRAISAL** element of the workflow container. This means that the binding is defined in the corresponding step for this element.

Workflow Template PdvAppraisal (PA-PD)

Definition

Workflow template for approving and carrying out follow-up steps for appraisals.

Please note that appraisals are created, held, and completed in the application, not in Workflow.

Workflow template: 1000105

Abbreviation: PdvAppraisal

Name: Appraisal

Structure

Workflow Container

The following information must be available during the workflow:

- the *(object) reference to the appraisal to be approved*
- the *name of the appraiser*

In order to process this information, the following elements must be available in the workflow container:

- The element **_WF_Initiator** for processing the appraiser is already defined in the workflow container in the standard system.
- The element **Appraisal** for processing the object reference was added to the existing standard system elements.

The information is available in the form of event parameters in the container of the triggering event, and has to be passed on from there using a binding to the workflow container.

Triggering Event of the Workflow Template

The event **COMPLETEDAPPRAISAL** (*Appraisal completed*) is defined for object type **BUS7026** (*Appraisal*) as the triggering event of the workflow template. This “link” between the event and the workflow template to be started has to be activated first in Customizing for this workflow template.

Binding of Event Parameters to the Workflow Container

The following binding is defined between the triggering event (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
_WF_Initiator	<-	_EVT_CREATOR
Appraisal	<-	_EVT_OBJECT

Integration

This workflow template describes a complex workflow that contains a number of subworkflows. See also:

[Workflow Definition Details \[Page 130\]](#)

[Workflow Template PdvAppEmpl \[Page 133\]](#)

Workflow Definition Details

Use

You use this workflow to pass an appraisal on for approval and follow-up processing.

Workflow Definition Details

Features

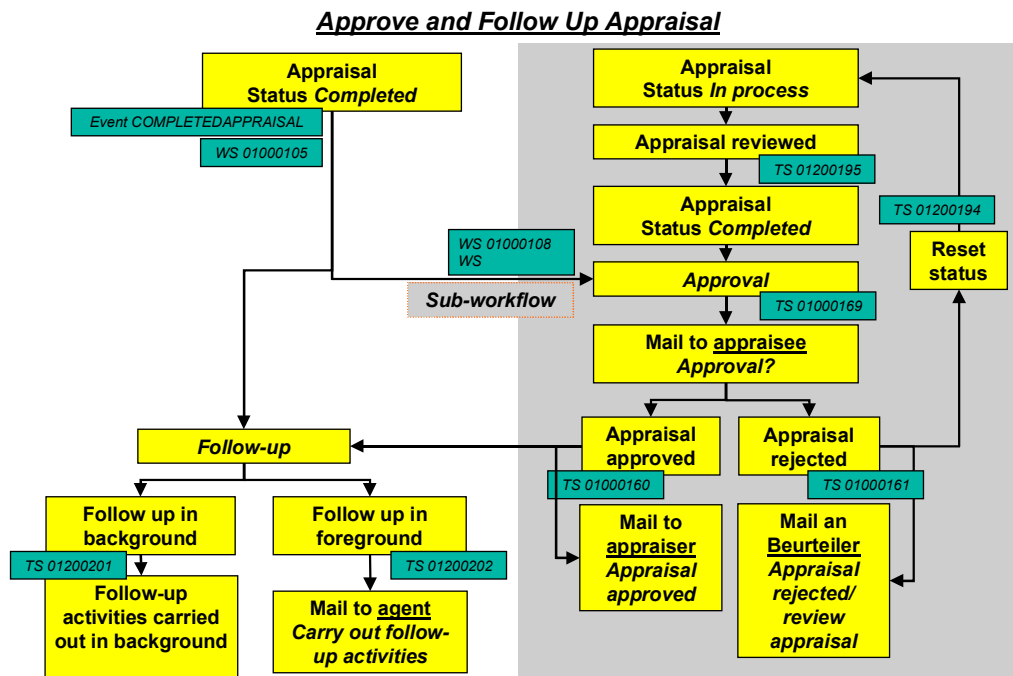
The following information might be useful with regard to the workflow definition for the workflow template **PdvAppraisal**. Please refer also to the definition in the system.

The workflow template comprises several other workflow templates (subflows).

- The triggering event is defined in the main workflow.
- Several alternative subflows might have been defined for the decision task.

The event that triggers the workflow is generated automatically when an appraisal is completed. For more information on the triggering event of the main workflow, see [Workflow Template PdvAppraisal \(PA-PD\) \[Page 129\]](#).

Scenario



1. An [appraiser \[Page 127\]](#) carries out an appraisal, and completes this appraisal. The completion of the appraisal causes a new object of the object type **BUS7026 (Appraisal)** to be created, and generates the event **COMPLETEDAPPRAISAL (Appraisal completed)** for this object type. This event triggers workflow template **WS01000105 (Appraisal)**. A multiple condition is used to determine the form of appraisal on which the appraisal is based.



The workflow can be linked to various forms of appraisal by means of subworkflows. The approval process is only triggered if a suitable workflow template exists and has been defined as a sub-workflow.

- If a suitable workflow template has not been defined for the form of appraisal in question, the workflow goes directly to follow-up processing (step 4).

Workflow Definition Details

- If the form of appraisal *Regular appraisal (supervisor appraises employee)* is used for the appraisal, workflow template **ws01000108** (*Appraisal for Employee*) is started.



Workflow template **ws01000108** ([Appraisal for Employee \[Page 133\]](#)) can be used as a copy template for creating other workflow templates that can be linked to other forms of appraisal (reverse appraisals, business event appraisals, and so on).

2. Role **01000016** (*Appraisee*) is used to determine the appraisee. The appraisal is then sent to this person, along with decision task **ts01000169** (*Approve decision*).
3. Depending on the appraisee's decision, either standard task **ts01000160** (*Approve appraisal*) or **ts01000161** (*Reject appraisal*) is executed. Role **01000017** (*Appraiser*) is used to determine the appraiser, who is informed of the appraisee's decision.

Appraisal approved	Appraisal rejected
If the appraisal was approved, an e-mail is sent to the appraiser to inform him/her of this, and the sub-workflow is completed.	<p>If the appraisal was rejected, standard task ts01200194 (<i>Reset Status of Appraisal</i>) is used to change the status of the appraisal back to <i>In process</i>. The appraiser is informed of this, and the appraisal is returned to the appraiser to be reviewed.</p> <div data-bbox="954 936 1006 987" data-label="Image"> </div> <p>If the status cannot be reset, the appraiser is informed that the appraisal has been rejected. This completes the sub-workflow and the main workflow.</p> <p>When standard task ts01200195 (<i>Change and Complete Appraisal</i>) is executed, this re-triggers decision task ts01000169 (<i>Approve Decision</i>).</p>

4. If you have specified that appraisals should be followed-up, the follow-up activities in question are carried out; otherwise, the workflow is ended once the appraisal has been approved.

Follow up in background	Follow up in foreground
Standard task ts01200201 (<i>Approve Appraisal</i>) is executed for follow-up activities to be performed in the background. This runs automatically, and cannot be canceled. No correspondence is triggered.	Standard task ts01200201 (<i>Approve Appraisal</i>) is executed for follow-up activities to be carried out in the foreground, and role 01200025 (Supervisor of Appraisee [Page 128]) is used to determine the correct agent, who is informed of this task. The agent can then carry out or cancel the follow-up activities.



The follow-up activities to be carried out (and how they are to be carried out) are determined by the appraisal model on which the appraisal is based.

Workflow Template PdvAppEmpl (PA-PD)

Result

The workflow is completed.

Workflow Template PdvAppEmpl (PA-PD)

Definition

Workflow template for deciding whether to approve appraisals.

Workflow template: 1000108

Abbreviation: PdvAppEmpl

Name: *Appraisal for Employee*

Structure

The (object) reference to the appraisal to be approved must be available during this workflow. In order to process the object reference, the element *Appraisal* must be available in the workflow container.

Integration

This workflow template is used as a copy template for creating other workflow templates. You can use it to create workflow templates for other forms of appraisal (reverse appraisals, business appraisals, course instructor appraisals, and so on) and define these as sub-workflows in workflow template **1000105** (*Appraisal*).

Preparations and Customizing (PA-PD)

Once you have carried out general customizing, which ensures that the Workflow System functions correctly, you need to customize the workflow template itself.



The information below is irrelevant if the workflow is only used for following up appraisals (that is, there is no approval process). Please note that the follow-up activities to be carried out (and the way in which they are to be processed) is determined by the appraisal model on which the appraisal in question is based. You define forms of appraisal in Customizing for *Personnel Development* (step: *Appraisal Systems* → [Edit Appraisals Catalog \[Ext.\]](#)).

Prerequisites

You have completed general Customizing for the SAP Business Workflow system.

Customizing the Workflow Definition

- In the workflow definition for workflow template **WS01000105** (*Appraisal*), you specify which form of appraisal triggers which sub-workflow. You define forms of appraisal in Customizing for *Personnel Development* (step: *Appraisal Systems* → [Edit Form of Appraisal \[Ext.\]](#)). There are no standard entries for forms of appraisal. Workflow template **WS01000108** (*Appraisal for Employee*) is only intended as an example, and has not been linked to any form of appraisal.

This also means that you will have to use the *Workflow Builder* to customize the workflow definition of workflow template **WS01000105** in the step entitled *Form of appraisal*.

You must add a new step to the multiple condition *Form of appraisal* for each form of appraisal for which you to use the approval workflow. The reference is derived from the ID of the form of appraisal. You can use the existing template to copy from.

- Depending on the way in which the approval workflow is to be used (that is, which appraisal models are to be used), you might have to change the roles for standard tasks **TS 1000169** ([Approve Decision \[Page 124\]](#)) and **TS 1200195** ([Change and Complete Appraisal \[Page 125\]](#)).
- Make sure that you are allowed to reset the status to *In process* for all of the appraisal models used in the approval process. If you do not, the entire workflow will be ended when the appraisal is rejected. You make these settings in Customizing for *Personnel Development* (step: *Appraisal Systems* → [Edit Appraisals Catalog \[Ext.\]](#)).

Working Through Task-Specific Customizing

In **Customizing** for the SAP Business Workflow system, the standard tasks **TS 1000169** (*Approve decision*), **TS 1200195** (*Change and Complete Appraisal*) and **TS 1200202** (*Follow Up an Appraisal in the Foreground*) have to be assigned to possible agents.

Procedure

1. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
2. Choose the step *Assign task to agent* in the *Personnel Development* component.
3. Classify standard tasks **1000169**, **1200195** and **1200202** as general tasks.

Linking Events to Event Receivers

The event **COMPLETEDAPPRAISAL** defined for object type **BUS7026** is the triggering event for the workflow template **1000105** and, consequently, is specified in the linkage table in the standard system.

To ensure that the workflow template really is started, you must **activate** the link between the triggering event and the workflow template as the receiver of the event. You do this in **Customizing** for SAP Business Workflow.

Connection to Application Functions (PA-PD)

Procedure

1. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
2. Activate the link for the workflow template *Appraisal* in the *SAP Business Workflow* application component

Alternatively, you can activate the link between the event and the event receiver by editing the workflow template `PdvAppraisal` directly.

Connection to Application Functions (PA-PD)

Prerequisites

You have made the settings described in the topic [Preparations and Customizing \(PA-PD\) \[Page 133\]](#).

Process Flow

The process flow of this workflow can be seen by looking at the process flow for the workflow for approving an appraisal.

Holding an Appraisal

A supervisor creates and holds an appraisal for an employee.



Both persons or users must be the holders of positions.

You can carry out the activities in question from the appraiser's *Appraisals where appraiser* subprofile.

Completing an Appraisal

The supervisor completes the appraisal. This means that the status of the appraisal is not *Completed*.



The appraisal process comprises several phases. During each of these phases, the appraisal itself has a specific status. The statuses *Approved* and *Rejected* can be assigned only by this workflow.

The completion of an appraisal is the *triggering event* of the workflow template for approving an appraisal.

Rejecting an Appraisal

The employee finds a work item in his/her integrated inbox. The appraisee can display the appraisal from the work item. This work item permits only the "approval" or "rejection" of the appraisal. The employee rejects the appraisal.

Changed Subprofile (PA-PD)



Theoretically, all employees (that is, possible agents) can approve their appraisal, although you can restrict this in Customizing. However, at runtime, only the appraisee for whom the appraisal was held (that is, the agent selected) should be able to decide. This is achieved by means of role resolution.

Reviewing an Appraisal

The supervisor finds a work item in his/her integrated inbox. The appraisal can be displayed from this work item. This work item permits the “reviewing” of the appraisal. The supervisor changes and completes the appraisal.

Approving an Appraisal

The employee finds a work item in his/her integrated inbox. The appraisee can display the appraisal from the work item. This work item permits only the “approval” or “rejection” of the appraisal. The employee approves the appraisal.

The appraiser receives information as to the appraisee’s approval in his or her integrated inbox.

Following Up an Appraisal

The supervisor finds a work item in his/her integrated inbox. This work item permits only the processing or cancellation of a follow-up processing activity. The supervisor carries out the follow-up processing activity.

The workflow is completed.

Changed Subprofile (PA-PD)

Purpose

This workflow describes how the [subprofiles \[Ext.\]](#) of objects are monitored.

If any changes are made to a subprofile, the “owner” of the subprofile (that is, the object for which the subprofile was created) is informed of this. The subprofile in question is also sent for the owner’s perusal. This workflow applies to all of the subprofiles in *Personnel Development* except for the subprofiles for appraisals (*Appraisals where appraiser*, *Appraisals where appraisee*). In the case of a [Requirements \[Ext.\]](#) subprofile, the holder(s) of the job or position in question are determined, and informed of the change. In this case, the current [profile matchup \[Ext.\]](#) comparing the requirements with the current qualifications is sent.

It is irrelevant how the changes occurred (changes made by an administrator or supervisor, for example, changes made using the [Employee Self-Service for qualifications \[Ext.\]](#), changes made by updating qualifications as a result of business event attendance or appraisals, and so on). All types of changes are taken into account (new, changed, or deleted relationships, and new, changed, or deleted notes).

You use this workflow if, for example, you want to:

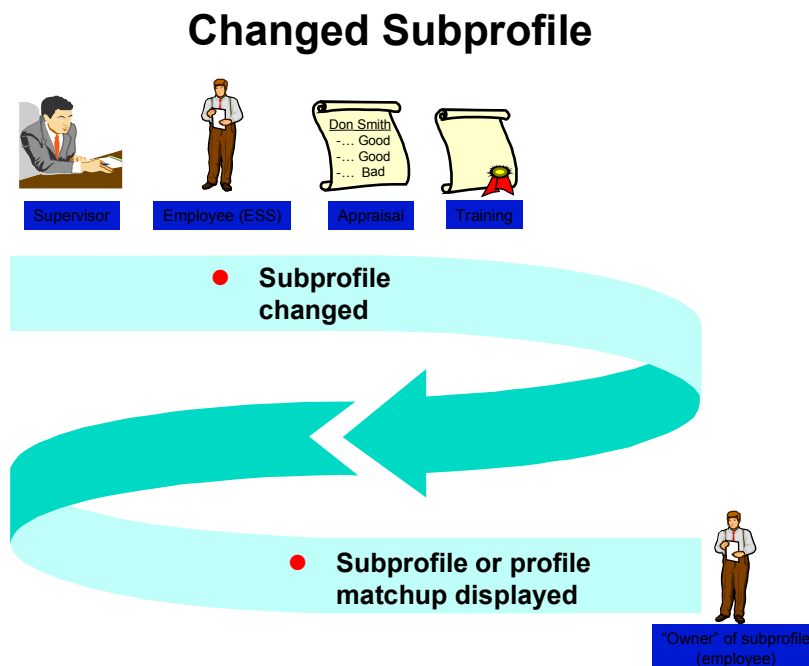
- Inform employee of changed requirements

Changed Subprofile (PA-PD)

- Inform employees of changes in their individual development plans
- Give notification that changes were made using the Employee Self-Service for qualifications

You can also use this workflow for other roles – for example, if you want to:

- Enable an employee's supervisor to monitor changes made using the Employee Self-Service for qualifications
- Inform supervisors of changes made to the requirements of jobs that form the basis for some positions in the supervisors area of responsibility
- Inform anyone interested or responsible of changes made to subprofiles

Process Flow

1. A subprofile is changed. The workflow is triggered automatically when this change is saved.
2. The "owner" of the subprofile, or holder of the position (job) in question is determined, and receives a work item with the changed subprofile or profile matchup, which he/she can view.

Result

The workflow ends when the work item is executed.

Technical Implementation (PA-PD)

Object Types

Object-based technology means that it is possible to create an interface between R/3 functionality and the Workflow System.

The information that follows is very technical. You only need to read it if you require detailed technical information on implementing workflow, or if you want to make enhancements of your own.

[Object type BUS7030 \[Page 138\]](#)

Standard Tasks

The standard tasks that come with the standard SAP System (one-step tasks) describe elementary business activities from an organizational point of view.

The [standard task PdvDispProfi \[Page 139\]](#), which is used here, is available in the form of a module, and can be used as it is (i.e. without being changed) in other workflow scenarios.

Roles

The role entitled [Employee \[Page 146\]](#) is used to determine the agent.

Workflow Templates

The business process flow has been provided as a workflow template. You will find the [workflow template PDVChanProfi \[Page 140\]](#) in your R/3 System.

Object Type BUS7030 (PA-PD)

Definition

A subprofile of an object is a business application object of type **BUS7030**.

To view technical data on the object type, go to the Business Object Repository (*Personnel management* → *Personnel development*).

Use

In the scenario, any changes to objects of this type are evaluated.

Structure

Key Fields

An object of this type is identified uniquely by means of the following fields:

- Plan version
- Object type

Standard Task PdvDispProfi (PA-PD)

- Object ID
- Profile maintenance view
- HR development subprofile number

Standard Task PdvDispProfi (PA-PD)

Definition

When this standard task is executed, either a profile matchup is displayed (if the changes were made to a *Requirements* subprofile) or the subprofile that was changed is displayed (applies to all other subprofiles except *Appraisals where appraiser* and *Appraisals where appraiser*).

Standard task: 1200191

Abbreviation: PDVDispProfi

Name: *HR Development Subprofile or Display Profile Matchup*

Structure

Method Referenced, Properties

Object type: BUS7030 (*development subprofile*)

Method: SHOWSUBPROFILEORPROFILEMATCHUP (*HR Subprofile or Display Profile Matchup*)

Agent Assignment

At runtime, this standard task is addressed to the “owner” of the changed subprofile in question using role resolution. In the case of a [Requirements \[Ext.\]](#) subprofile, the holder(s) of the job or position in question are determined. Since the employee in question can be any employee, the task must be classified as a *general task*.

You can also use another role here (for example, **157 Manager**) if, for example, you wish to inform the employee’s supervisor, and not the employee, of the changes made to the subprofile.

Role of Employee (PA-PD)

Definition

Role used for determining the “owner” of a subprofile.

Role: 01200024

Abbreviation: PDVEMPLOYEE

Name: *Employee*

Structure

Role Parameters

The following role parameter is passed on:

- **Org_EMPLOYEE** (reference field **RHOBJECTS-OBJECT**); contains the concatenated keys (type and ID) of organizational objects

The role parameter exists as an element in the role parameter container.

Binding Definition

The role is used to find the correct “owner” of a subprofile (starting out from the initiator of the workflow). The owner is determined directly for all subprofiles except *Requirements*. In the case of requirements, the system reads the position and then its holder; in the case of jobs, the corresponding positions and their holders.

The object ID of the changed subprofile is stored in a 14-character field (reference **RHOBJECTS-OBJECT**) in the Element **_WF_Initiator** of the workflow container. This means that the binding is defined in the corresponding step for the element **ORG_USER**.

Integration

Preparations in the Organizational Plan

If the role is to be resolved correctly for *Requirements* subprofiles, each possible agent must hold a position.

Workflow Template PDVChanProfi (PA-PD)

Definition

Workflow template for monitoring changes made to Personnel Development subprofiles.

Workflow template: 1200124

Abbreviation: PDVChanProfi

Name: *Changed HR Profile*

Structure

Workflow Container

The following information must be available during the workflow:

- the *(object) reference to the changed subprofile*
- the *agent*

In order to process this information, the following elements must be available in the workflow container:

Workflow Template PDVChanProfi (PA-PD)

- The element **_WF_Initiator** for processing the agent is already defined in the workflow container in the standard system.
- The element **HRDevelopSubProfile** for processing the object reference was added to the existing standard system elements.

The information is available in the form of event parameters in the container of the triggering event, and has to be passed on from there using a binding to the workflow container.

Triggering Event of the Workflow Template

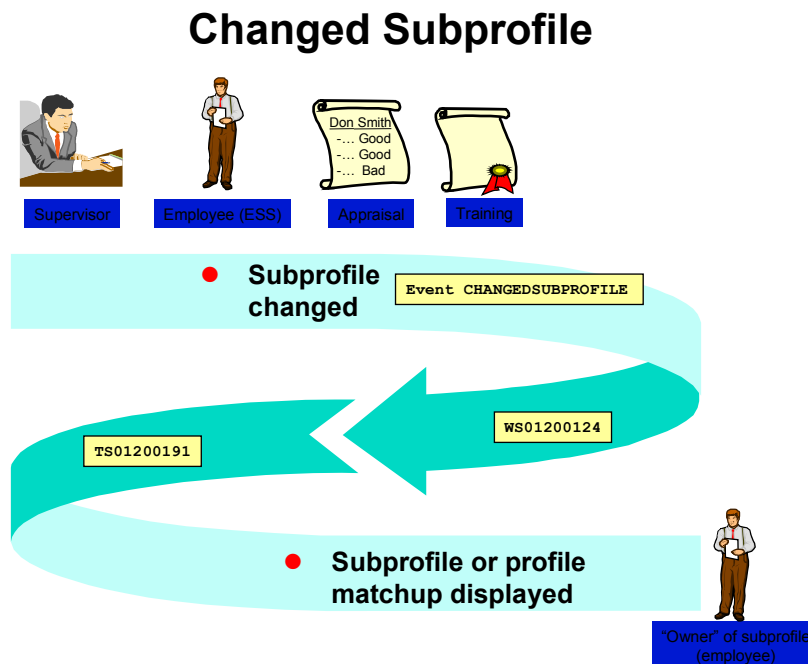
The event **CHANGEDSUBPROFILE** (*Changed HR Development Subprofile*) is defined for object type **BUS7030** (*Development Subprofile*) as the triggering event of the workflow template. This “link” between the event and the workflow template to be started has to be activated first in Customizing for this workflow template.

Binding of Event Parameters to the Workflow Container

The following binding is defined between the triggering event (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
_WF_Initiator	<-	_Evt_Creator
HRDevelopSubProfile	<-	_Evt_Object

Process Flow



Preparations and Customizing (PA-PD)

1. A subprofile is changed. When the change is saved, this generates the event **CHANGEDSUBPROFILE** (*Changed HR Development Subprofile*) for an object of the object type **BUS7030** (*Development Subprofile*). This event triggers workflow template **1200124** (*Changed HR Profile*).
2. Standard task **1200191** (*Display HR Development Subprofile or Profile Matchup*) uses role **01200024** (*Employee*) to determine the owner of the subprofile or the holder of the position (job) in question, and sends a work item with the changed subprofile or a profile matchup to this person for his/her perusal.

Result

The workflow ends when the work item is executed.

Preparations and Customizing (PA-PD)

Once you have carried out general customizing, which ensures that the Workflow System functions correctly, you need to customize the workflow template itself.

Prerequisites

You have completed general Customizing for the SAP Business Workflow system.

Setting Up the Organizational Plan

The agents of work items for changed *Requirements* subprofiles must be assigned to positions. See [Organizational Plan \[Ext.\]](#).

Working Through Task-Specific Customizing

In **Customizing** for the SAP Business Workflow system, the standard task **1200191** ([Display HR Development Subprofile or Profile Matchup \[Page 139\]](#).) has to be assigned to possible agents.

Procedure

4. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
5. Choose the step *Assign task to agent* in the *Personnel Development* component.
6. Classify standard task **1200191** as a general task.

Linking Events to Event Receivers

Linking Events to Event Receivers

The event **CHANGEDSUBPROFILE** defined for object type **BUS7030** is the triggering event for the workflow template **1200124** and, consequently, is specified in the linkage table in the standard system.

To ensure that the workflow template really is started, you must **activate** the link between the triggering event and the workflow template as the receiver of the event. You do this in **Customizing** for SAP Business Workflow.

Procedure

3. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
4. Activate the link for the workflow template *Changed HR Subprofile* in the *Personnel Development* application component.

Alternatively, you can activate the link between the event and the event receiver by editing the workflow template **PDVChanProfi** directly.

Profile Matchup for Transfer to Another Position (PA-PD)

Purpose

When an employee is transferred to another position, this workflow is used to inform the employee of the requirements of the new position.

Once the employee has been transferred, a profile matchup is run to compare the employee's qualifications with the requirements of the new position. If the employee does not fulfill certain requirements, the profile matchup can generate training proposals that will provide the employee with the missing qualifications.

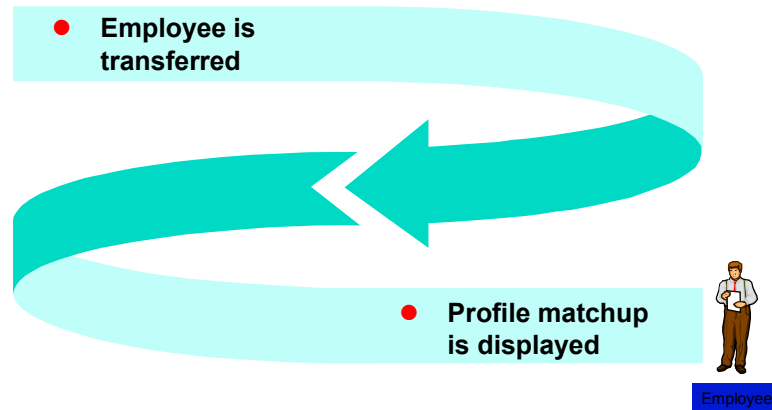
It does not matter who or what causes the transfer to take place (the employee's supervisor, for example, or a job rotation measure in an [individual development plan \[Ext.\]](#), etc.).

You use this workflow if, for example, you want to:

- Ensure that employees know what the requirements of their new positions are
- Ensure that employees are aware of any requirements that they do not meet
- Plan possible development measures by generating training proposals

Process Flow

Profile Matchup for Transfer to Another Position



1. An employee is transferred to another position. The workflow is triggered automatically when the employee is transferred.
2. A work item is sent to the employee. This work item contains a profile matchup where the requirements of the new position are compared with the employee's current qualifications.

Result

The workflow ends when the work item is executed.

Technical Implementation (PA-PD)

Object Types

Object-based technology means that it is possible to create an interface between R/3 functionality and the Workflow System.

The information that follows is very technical. You only need to read it if you require detailed technical information on implementing workflow, or if you want to make enhancements of your own.

[Object Type PDOTYPE_SH \[Page 145\]](#)

Object Type PDOTYPE_SH (PA-PD)**Standard Tasks**

The standard tasks that come with the standard SAP System (one-step tasks) describe elementary business activities from an organizational point of view.

The [standard task PdvPosTransf \[Page 145\]](#), which is used here, is available in the form of a module, and can be used as it is (i.e. without being changed) in other workflow scenarios.

Roles

The role entitled [Employee \[Page 146\]](#) is used to determine the agent.

Workflow Templates

The business process flow has been provided as a workflow template. You will find the [workflow template PdvTransfer \[Page 147\]](#) in your R/3 System.

Object Type PDOTYPE_SH (PA-PD)**Definition**

A position is a business application object of type `PDOTYPE_SH`.

To view technical data on the object type, go to the Business Object Repository (*Personnel management* → *Organizational Plan* → *Staffing*).

Use

In the scenario, the requirements of an object of this type are evaluated.

Structure**Key Fields**

An object of this type is identified uniquely by means of the following fields:

- Plan version
- Object ID

Standard Task PdvPosTransf (PA-PD)**Definition**

When this standard task is executed, a profile matchup is displayed comparing the employee's qualifications with the requirements of the employee's new position.

Standard task: 1200223

Abbreviation: PDVPosTransf

Name: *Transfer a person to another position*

Structure

Method Referenced, Properties

Object type: PDOTYPE_SH (*Position HR*)

Method: SHOWDEFIZITPROFILEMATCHUP (*Profile Matchup*)

Agent Assignment

At runtime, this standard task is addressed to the employee who has been transferred – this is achieved using role resolution. Since the employee in question can be any employee, the task must be classified as a *general task*.

Role of Employee (PA-PD)

Definition

Role used for determining the “owner” of a subprofile.

Role: 01200024

Abbreviation: PDVEMPLOYEE

Name: *Employee*

Structure

Role Parameters

The following role parameter is passed on:

- **Org_EMPLOYEE** (reference field RHOBJECTS-OBJECT); contains the concatenated keys (type and ID) of organizational objects

The role parameter exists as an element in the role parameter container.

Binding Definition

The role is used to find the correct “owner” of a subprofile (starting out from the initiator of the workflow). The owner is determined directly for all subprofiles except *Requirements*. In the case of requirements, the system reads the position and then its holder; in the case of jobs, the corresponding positions and their holders.

The object ID of the changed subprofile is stored in a 14-character field (reference RHOBJECTS-OBJECT) in the Element **_WF_Initiator** of the workflow container. This means that the binding is defined in the corresponding step for the element **ORG_USER**.

Workflow Template PdvTransfer (PA-PD)

Integration

Preparations in the Organizational Plan

If the role is to be resolved correctly for *Requirements* subprofiles, each possible agent must hold a position.

Workflow Template PdvTransfer (PA-PD)

Definition

When an employee is transferred to another position, this workflow template is used to inform the employee of the requirements of the new position.

Workflow template: 1200145

Abbreviation: PdvTransfer

Name: *Transfer to Another Position*

Structure

Workflow Container

The following information must be available during the workflow:

- the *(object) reference to the position*
- the *object ID of the holder of the position*
- the *object type of the holder of the position*

In order to process this information, the following elements must be available in the workflow container:

- The element **HRChangedPosition** for processing the object reference was added to the existing standard system elements.
- The element **TypeOfRelatedEmployee** for processing the object type of the transferred employee was added to the existing standard system elements.
- The element **IDOfRelatedEmployee** for processing the object ID of the transferred employee was added to the existing standard system elements.

The information is available in the form of event parameters in the container of the triggering event, and has to be passed on from there using a binding to the workflow container.

Triggering Event of the Workflow Template

The event **POSTTRANSFER** (*Transfer to Another Position*) is defined for object type **PDOTYPE_SH** (*Position HR*) as the triggering event of the workflow template. This “link” between the event and the workflow template to be started has to be activated first in Customizing for this workflow template.

Workflow Template PdvTransfer (PA-PD)

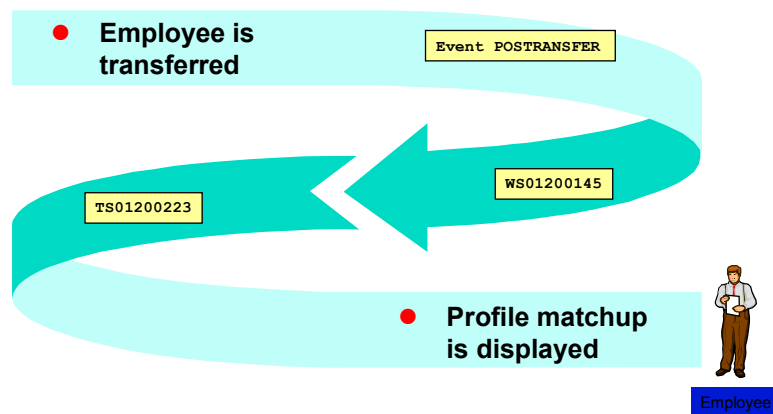
Binding of Event Parameters to the Workflow Container

The following binding is defined between the triggering event (event parameter container) and the workflow container:

Workflow Container		Event Parameter Container
HRChangedPosition	<-	_Evt_Object
TypeOfRelatedEmployee	<-	TYPEOFRELATEDEMPLOYEE
IDOfRelatedEmployee	<-	IDOFRELATEDEMPLOYEE

Process Flow

Profile Matchup for Transfer to Another Position



1. An employee is transferred to another position. When the transfer is made, this generates the event **POSTTRANSFER** (*Transfer to Another Position*) for an object of the object type **PDOTYPE_SH** (*Position HR*). This event triggers workflow template **1200145** (*Transfer to Another Position*).
2. Standard task **1200223** (*Transfer a person to another position*) uses role **01200024** (*Employee*) to determine the holder of the position, and sends a work item containing the profile matchup to this person for his/her perusal.

Result

The workflow ends when the work item is executed.

Preparations and Customizing (PA-PD)

Preparations and Customizing (PA-PD)

Once you have carried out general customizing, which ensures that the Workflow System functions correctly, you need to customize the workflow template itself.

Prerequisites

You have completed general Customizing for the SAP Business Workflow system.

Working Through Task-Specific Customizing

In **Customizing** for the SAP Business Workflow system, the standard task 1200223 (*Transfer a person to another position*) has to be assigned to possible agents.

Procedure

7. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
8. Choose the step *Assign task to agent* in the *Personnel Development* component.
9. Classify standard task 1200223 as a general task.

Linking Events to Event Receivers

The event **POSTTRANSFER** defined for object type **PDOTYPE_SH** is the triggering event for the workflow template **1200145** and, consequently, is specified in the linkage table in the standard system.

To ensure that the workflow template really is started, you must **activate** the link between the triggering event and the workflow template as the receiver of the event. You do this in **Customizing** for SAP Business Workflow.

Procedure

5. In Customizing for *SAP Business Workflow*, execute the activity *Perform Task-Specific Customizing*.
6. Activate the link for the workflow template *Transfer to Another Position* in the *Personnel Development* application component.

Alternatively, you can activate the link between the event and the event receiver by editing the workflow template **PdvTransfer** directly.

